

Projecte de Fi de Carrera  
**Enginyer Industrial**

**Identificació i control d'un motor  
de tipus *voice coil***

**MEMÒRIA**

**Autor:** Pablo Segovia Castillo  
**Director:** Vicenç Puig Cayuela  
**Convocatòria:** Abril 2015



Escola Tècnica Superior  
d'Enginyeria Industrial de Barcelona





## Resum

El present treball té dos objectius principals: el primer d'ells és el d'identificar el model de dues plantes - en aquest cas, dos motors de corrent continu - emprant diverses tècniques d'ajust a partir de dades experimentals; en segon lloc, es volen dissenyar estructures de control adequades sobre dites plantes, de forma que es satisfacin les especificacions demanades. En relació amb aquest darrer punt, es vol avaluar quin és el límit de prestacions dels motors; és a dir, quines són les especificacions màximes que se'ls hi poden demanar abans que aquests comencin a comportar-se de forma no esperada.

El primer capítol descriu l'equipament amb què es treballarà al llarg del projecte, format per aquests dos motors: d'una banda, un motor de tipus *voice coil* proporcionat per l'empresa *Ingenia Motion Control*; de l'altra, un dels exemplars que es troben al Laboratori de Control de l'ETSEIB.

El segon capítol descriu les diverses tècniques d'identificació de models que es fan servir per a caracteritzar-ne els paràmetres, i les aplica emprant jocs de dades del motor d'*Ingenia* i també amb dades generades per experimentació sobre el motor del Laboratori de Control. Als annexos es pot trobar informació detallada per a poder reproduir els experiments.

El tercer capítol comprèn el disseny de controladors de les plantes en base a l'especificació en *bandwidth*, i s'hi detallen les estructures de control PID, I+PD i RST estudiades en el present treball. Es para atenció al fenomen de la saturació i es desenvolupen codis de caràcter general que, a partir del *bandwidth* desitjat, proporcionen els guanys del controlador. Finalment, es presenten resultats en simulació i experimentals per a ambdós motors; pel que fa al motor d'*Ingenia*, s'observen diferències que plantegen les bases d'un nou projecte, que no forma part de l'abast del present.



# Índex

<b>Resum</b>	<b>3</b>
<b>Índex</b>	<b>7</b>
<b>Prefaci</b>	<b>9</b>
<b>Introducció</b>	<b>11</b>
<b>1 Descripció de l'equipament</b>	<b>13</b>
1.1 Motor d' <i>Ingenia</i> . . . . .	13
1.1.1 Motor . . . . .	13
1.1.2 Controlador . . . . .	15
1.1.3 Arquitectura del conjunt . . . . .	16
1.2 Motor del Laboratori de Control . . . . .	17
<b>2 Identificació</b>	<b>19</b>
2.1 Deducció de l'estructura de la planta . . . . .	19
2.2 Definició dels experiments i senyal multisinus . . . . .	21
2.3 Mètodes d'ajust . . . . .	22
2.3.1 <i>Identification Toolbox</i> . . . . .	22
2.3.2 <i>Output-Error</i> . . . . .	24
2.4 Resultats de la identificació . . . . .	28
2.4.1 Verificació del mètode en simulació . . . . .	31
2.4.2 Resultats experimentals sobre el motor del Laboratori (llaç obert) . . . . .	35
2.4.3 Resultats experimentals sobre el motor d' <i>Ingenia</i> . . . . .	38
2.4.4 Resum dels resultats . . . . .	47

2.5	Conclusions sobre la identificació . . . . .	49
<b>3</b>	<b>Control</b>	<b>51</b>
3.1	Definició de les especificacions . . . . .	51
3.2	Estratègies de control i estructures emprades . . . . .	52
3.2.1	Disseny en temps continu i discretització . . . . .	52
3.2.2	Disseny en temps discret . . . . .	56
3.3	Determinació dels pols a partir del <i>bandwidth</i> . . . . .	59
3.3.1	Mètode general en temps continu . . . . .	60
3.3.2	Aplicació del resultat al cas RST . . . . .	61
3.4	Assignació de pols a través del codi desenvolupat . . . . .	62
3.4.1	Controladors PID i I-PD . . . . .	62
3.4.2	Controlador RST . . . . .	63
3.5	Límit de prestacions dels motors . . . . .	64
3.6	Resultats del control . . . . .	67
3.6.1	Motor del Laboratori . . . . .	68
3.6.2	Motor d' <i>Ingenia</i> . . . . .	75
3.6.3	Resum dels resultats . . . . .	81
3.7	Conclusions sobre el control . . . . .	82
	<b>Conclusions</b>	<b>83</b>
	<b>Bibliografia</b>	<b>86</b>
	<b>A Pressupost</b>	<b>87</b>
	<b>B Estudi d'impacte ambiental</b>	<b>89</b>
	<b>C Codis</b>	<b>91</b>
C.1	Codis per a la identificació . . . . .	91
C.1.1	Generació del senyal multisinus . . . . .	91
C.1.2	Codi d'ajust amb estructura OE . . . . .	92
C.2	Codis per al control . . . . .	94
C.2.1	Solució dels controladors PID i I-PD continus . . . . .	94
C.2.2	Solució del controlador RST discret . . . . .	95
C.2.3	Codi per a trobar el <i>bandwidth</i> límit . . . . .	97

---

<b>D Guia d'usuari del motor d'<i>Ingenia</i></b>	<b>101</b>
D.1 Generació de dades per a la identificació . . . . .	101
D.2 Aplicació dels controladors dissenyats . . . . .	102
<b>E Fitxa del motor de <i>Bei Kimco Magnetics</i></b>	<b>103</b>
<b>F Fitxa del <i>Pluto DC Servo Drive</i></b>	<b>104</b>
<b>G Fitxa del motor del Laboratori</b>	<b>105</b>





## Prefaci

El treball realitzat parteix de la voluntat d'aprofundir en el món de la recerca en els àmbits de l'automàtica i del control, aplicant els coneixements adquirits al llarg de la intensificació d'Automàtica. A l'hora de fer la tria del *Projecte de Fi de Carrera* (PFC), però, també es desitjava que el treball fos *hands-on*, que permetés *fer d'enginyer* en el seu sentit més literal, treballant sobre sistemes físics. És per això que es va considerar que la millor forma de desenvolupar el PFC seria a través de la participació en un projecte real d'enginyeria, tot combinant desenvolupaments teòrics amb aplicacions pràctiques.

Aquesta oportunitat es va presentar en conèixer la possibilitat de col·laborar amb l'empresa *Ingenia Motion Control*, especialitzada en el disseny, desenvolupament i fabricació de sistemes orientats a l'àmbit del *motion control*. Poder desenvolupar el PFC tot col·laborant amb *Ingenia* oferia la possibilitat de conèixer el funcionament d'una empresa i el cicle de vida d'un projecte, amb totes les seves implicacions, alhora que permetia continuar la formació en el camp de l'automàtica, des dels punts de vista teòric i pràctic.



# Introducció

Com ja s'ha comentat en el prefaci, el present treball s'elabora en el marc d'un projecte real, que té com a objectiu dissenyar els controls dels llaços de corrent i de posició d'un motor de corrent continu. En el moment d'emprendre la col·laboració, el llaç de corrent ja estava satisfactòriament resolt, restant només per dissenyar el control del llaç de posició. Per tant, el PFC només versa sobre el llaç de control de la posició.

L'objectiu principal d'aquest treball és dissenyar una o diverses estructures de control de posició que permetin assolir les especificacions desitjades per a aquest motor. Per tal que l'acció de control actuï tal i com s'ha predit, cal identificar prèviament el model matemàtic del motor: si aquest pas no es fa adequadament, la resposta real del motor diferirà de la resposta obtinguda en simulació. Es considera d'interès desenvolupar codis de caràcter general, aplicables en altres contextos, de forma que el treball realitzat pugui ser estès a un major nombre de casos.

Directament relacionat amb el punt anterior, hom pot plantejar-se quin és el límit de prestacions que ofereix el motor; és a dir, quines són les especificacions màximes que poden exigir-se al motor abans que aquest sature. Estudiar acuradament aquest fenomen és crucial, ja que introdueix efectes no desitjats en la resposta del motor.

La metodologia seguida parteix d'uns resultats obtinguts a través de la simulació de construccions teòriques, els quals posteriorment es validen repetint els experiments sobre els motors. Resulta de gran importància fer aquesta validació, ja que, en ocasions, dites construccions teòriques no són capaces de capturar completament la naturalesa dels elements físics, i és possible que ambdós resultats difereixin.

Cal comentar que en el decurs d'aquest projecte es treballa sobre dos motors diferents: d'una banda, un motor de tipus *voice coil* proporcionat per l'empresa *Ingenia Motion Control*; de l'altra, un dels exemplars del laboratori de Control de l'ETSEIB. Procedint d'aquesta forma es pot garantir en major mesura que els resultats obtinguts són correctes.



## Capítol 1

# Descripció de l'equipament

En aquest capítol es descriuen les plantes que es fan servir per a realitzar els estudis d'identificació i control del present treball. Se'n presenta una visió molt general, i s'inclouen les fitxes tècniques i altres recursos als annexos per tal d'ampliar aquesta informació.

### 1.1 Motor d'*Ingenia*

El sistema anomenat de forma reiterada com a *motor d'Ingenia* consta d'un motor de tipus *voice coil* de corrent continu de l'empresa *Bei Kimco Magnetics* i d'un controlador de llaç tancat d'alt rendiment de la gamma *Pluto*, desenvolupat per *Ingenia Motion Control*. El conjunt s'ha d'alimentar a 24 volts per tal que funcioni correctament. A més, es fa ús de la seva sortida Micro-B USB per a establir el canal de comunicació amb el PC.

A continuació es tracten per separat el motor i el controlador, i finalment es presenta la relació entre ambdós elements.

#### 1.1.1 Motor

En el present cas d'estudi, *Ingenia* treballa amb un motor de l'empresa *Bei Kimco Magnetics*; concretament, amb el LAS16-23-00A, un petit motor lineal de corrent continu de tipus encapsulat. Se'n pot trobar la seva fitxa de producte a l'annex E. El seu aspecte és el següent:



Figura 1.1: Detall del motor LAS16-23-00A de *Bei Kimco Magnetics*

El principi d'actuació d'un motor *voice coil* [5] és el d'emprar un camp magnètic permanent sobre una bobina a la qual se li aplica un corrent elèctric, de forma que es produeix una força proporcional al corrent aplicat a la bobina. Aquests actuadors són àmpliament utilitzats - [2] i [5] - en aplicacions lineals i rotatives que requereixen respostes en freqüències elevades, gran acceleració, velocitat de resposta elevada i control amb gran precisió.

El seu funcionament [2] és governat per la llei de Lorentz, que enuncia que si un conductor pel qual circula un corrent es situa en el si d'un camp magnètic, rebrà l'acció d'una força proporcional al camp magnètic  $B$  i al corrent elèctric  $I$ . La direcció d'aquesta força depèn també de les direccions de  $B$  i  $I$ .

La configuració més senzilla d'un motor *voice coil* lineal és [5] la d'una bobina de coure situada dins d'un camp magnètic amb orientació radial. Aquest camp magnètic és creat per la inserció d'imants permanents en l'interior d'un cilindre de material ferromagnètic, disposats de manera que els imants que s'encaren a la bobina presenten tots la mateixa polaritat. Per a completar aquest circuit magnètic es situa un nucli de material ferromagnètic al llarg de l'eix central de la bobina i unit en el seu extrem a la configuració d'imants permanents. La següent imatge facilita la comprensió d'aquesta disposició:

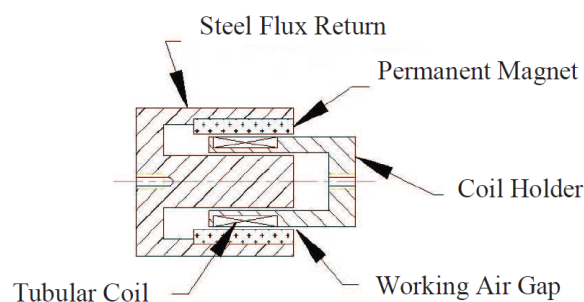


Figura 1.2: Configuració interna d'un motor *voice coil*



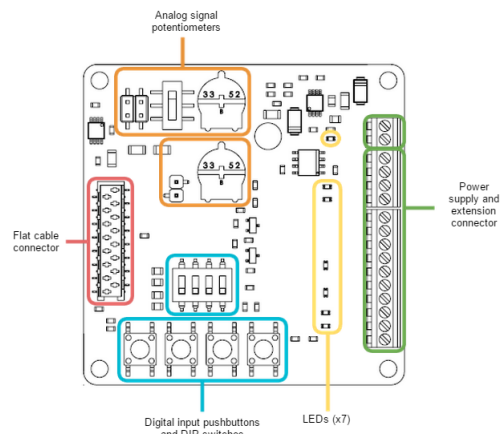


Figura 1.4: Detall de les connexions del *Pluto Servo Drive*

### 1.1.3 Arquitectura del conjunt

El controlador *Pluto* inclou dos llaços de control: un d'ells el fa des d'un punt de vista elèctric - corrent; i l'altre, des d'un punt de vista mecànic - posició. Si bé s'ha comentat que el treball només es centra en el llaç de posició, es considera interessant mostrar com estan relacionats els dos llaços. La següent figura mostra la connexió entre aquests dos llaços i la resta d'elements que conformen el sistema d'estudi:

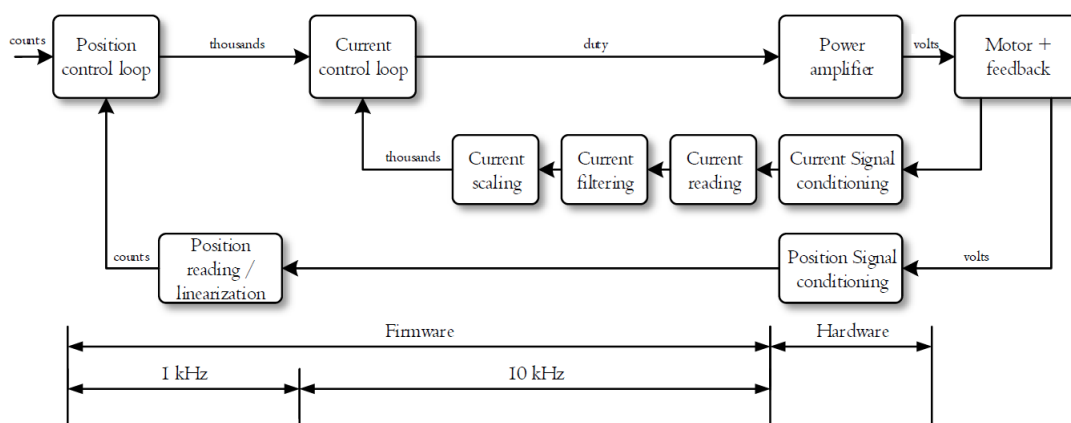


Figura 1.5: Diagrama de blocs del conjunt

La següent imatge mostra la visió del conjunt motor-controlador degudament connexionada



a l'ordinador. D'aquesta manera, es poden fer, per exemple, experiments per a adquirir dades sobre les quals fer la identificació del model, o bé provar els controladors un cop aquests ja estiguin dissenyats.

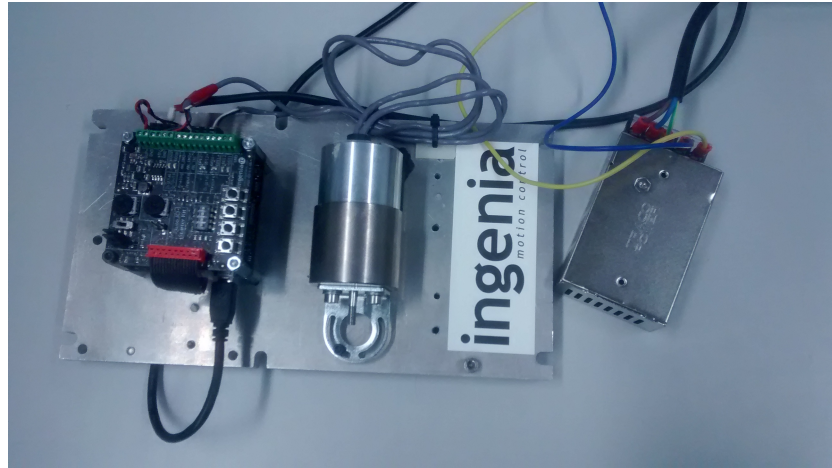


Figura 1.6: Conjunt motor-controlador d'*Ingenia*

## 1.2 Motor del Laboratori de Control

L'altra planta que s'estudia en el present treball és un dels exemplars que es troben en el Laboratori de Control del pis 2 de l'ETSEIB i és desenvolupada per l'empresa *LJ Create Ltd*. Cal tenir present que el motor de corrent continu que incorpora per a l'estudi és de tipus rotatori, a diferència del motor presentat en l'apartat 1.1.1, que és de tipus lineal, si bé l'estudi es podrà realitzar de la mateixa manera.

Aquest conjunt educatiu està dissenyat per a poder provar gran multitud de controladors - tant en temps continu com en temps discret - sobre un motor de corrent continu. Disposa d'*encoders* Gray absolut i incremental que mesuren la posició, i d'un tacòmetre per a mesurar la velocitat. A més, es pot treballar amb un *software* propi per a mostrar i analitzar els resultats del control, tot i que des del departament d'ESAI s'ha desenvolupat una interfície amb el *Simulink* per a establir la connexió entre el PC i la planta.

La següent imatge mostra una visió global d'aquesta planta:

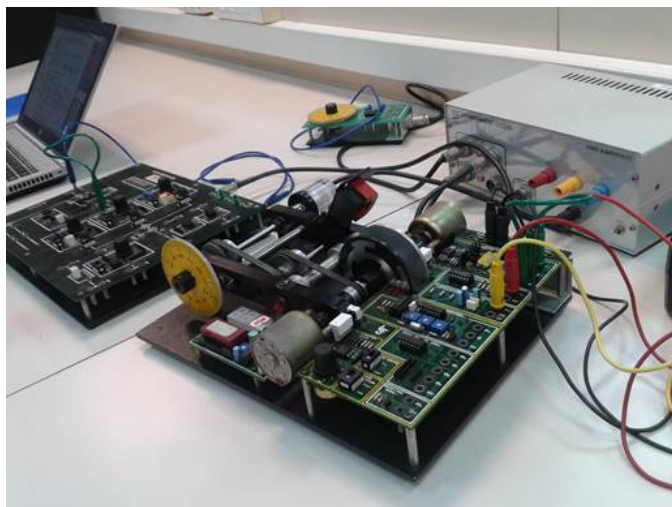


Figura 1.7: Planta del Laboratori de Control

Per a més informació sobre aquesta planta es pot consultar el guió de pràctiques [4] de l'assignatura de Control amb Computador. Allà s'hi poden trobar totes les figures i esquemes necessaris per a una comprensió més profunda d'aquesta planta. A més, la fitxa d'aquest producte és a l'annex G.

## Capítol 2

# Identificació

El segon capítol està dedicat a la identificació dels paràmetres de les plantes per obtenir-ne models matemàtics que les representin de la forma més versemblant possible, de forma que l'acció de control - estudiada en el capítol 3 - que s'hi apliqui sigui efectiva.

El capítol està estructurat de la següent manera: en primer lloc, s'empren les equacions físiques que descriuen el motor per a deduir el model de la planta. A continuació es defineixen de forma general els experiments i es presenten els mètodes d'ajust que es fan servir per identificar els models. Finalment es presenten els resultats obtinguts i les conclusions que se'n poden extreure.

En aquest primer capítol encara no s'interactua de forma física amb el motor d'*Ingenia*, sinó que els *datasets* són directament proporcionats per l'empresa. En l'annex D.1, però, es descriu el procediment que permet reproduir els experiments per tal d'obtenir les dades, les quals constitueixen el punt de partida del present capítol.

### 2.1 Deducció de l'estructura de la planta

Davant d'un problema d'identificació de models a partir de les mesures d'entrada i sortida de la planta, el primer que cal fer és trobar quina és l'estructura - ordres de la funció - idònia sobre la qual fer l'ajust. Si bé en ocasions aquest aspecte pot ser una incògnita i no hi ha altra solució que la prova i error, en el present cas d'estudi no succeeix això. Tal i com es defineix en el capítol 1 de [14], si es parteix del model mecànic d'un motor i únicament es consideren el moment d'inèrcia de l'eix  $J$  i la fricció viscosa  $B$ , l'expressió que relaciona el parell aplicat sobre l'eix i la velocitat angular és:

$$\Gamma(t) = K_t i(t) = J \frac{d\omega(t)}{dt} + B\omega(t) \quad (2.1)$$

on  $K_t$  és la constant de parell, que relaciona el corrent que circula amb el parell generat.

El motor d'*Ingenia* no disposa de sensor de velocitat - el motor del Laboratori sí que disposa d'un tacòmetre - sinó que únicament pot mesurar la posició angular. La relació entre posició i velocitat angular és:

$$\omega(t) = \frac{d\theta(t)}{dt} \quad (2.2)$$

Substituint l'equació (Eq. 2.2) en (Eq. 2.1) es té:

$$K_t i(t) = J \frac{d^2\theta(t)}{dt^2} + B \frac{d\theta(t)}{dt} \quad (2.3)$$

Aplicant la transformada de Laplace a l'equació (Eq. 2.3) s'obté:

$$K_t i(s) = J\theta(s)s^2 + B\theta(s)s \quad (2.4)$$

Anomenant  $G_{motor}(s)$  la funció de transferència que relaciona la posició angular amb el corrent que circula pel motor, aquesta es pot obtenir fàcilment de l'equació (Eq. 2.4).

$$G_{motor}(s) = \frac{\theta(s)}{i(s)} = \frac{\frac{K_t}{B}}{s \left( \frac{J}{B}s + 1 \right)} \quad (2.5)$$

on es veu clarament que el model resultant és un sistema amb cap zero i dos pols - un d'ells és un integrador.

L'equació (Eq. 2.5) s'usarà per a fer l'ajust del model del motor d'*Ingenia*, ja que no disposa d'un sensor de velocitat i sí d'un de posició. Es veurà a l'apartat 2.3 que caldrà un procés previ per a ajustar correctament els paràmetres del motor d'*Ingenia* precisament pel fet de no disposar de sensor de velocitat.

Per al cas del motor del Laboratori de Control, com que es disposa de tacòmetre, no és necessari el pas de velocitat angular a posició angular descrit en (Eq. 2.2), de forma que el model final que es farà servir és

$$G_{motor}(s) = \frac{\omega(s)}{i(s)} = \frac{\frac{K_t}{B}}{\frac{J}{B}s + 1} \quad (2.6)$$

Podria semblar que aquesta feina d'identificació del motor és innecessària si ja es té una

expressió per a la planta, on la única feina restant és la de substituir els termes  $K_t$ ,  $B$  i  $J$ . Partint del fet que no es disposa d'aquests valors, ja que normalment els fabricants no els faciliten, també cal tenir en compte la inèrcia pròpia de la càrrega que s'acobli a l'eix del motor; el seu efecte es pot modelar com una pertorbació sobre el sistema, la qual ha de ser corregida pel llaç de control. Per tant, es prescindeix d'aquests paràmetres físics i es transforma el model físic en un model orientat a control, essent

$$G(s) = \frac{K}{s(\tau s + 1)} \quad (2.7)$$

el model orientat a control del motor d'*Ingenia*, i

$$G(s) = \frac{K}{\tau s + 1} \quad (2.8)$$

el model orientat a control del motor del Laboratori de Control.

Per tant, l'objectiu serà trobar els valors de  $K$  i  $\tau$  que facin que el model obtingut sigui el que millor ajusti les dades experimentals.

## 2.2 Definició dels experiments i senyal multisinus

Per a fer la identificació del model del motor, cal aplicar un senyal d'entrada i registrar-ne la sortida corresponent: a partir d'aquests senyals d'entrada i sortida es realitzarà l'ajust per trobar el model del motor. El senyal triat s'obté a partir d'una combinació de senyals sinusoidal - anomenat *multisinus* a partir d'ara - que cobreix l'espectre de freqüències necessari per a l'estudi.

El codi desenvolupat per a obtenir el senyal d'entrada és el *IngeniaMCmsin*, que es pot trobar a l'annex C.1.1. Cal especificar el rang de freqüències que es desitgen contemplar, i a continuació es computa la operació *msinclip2*, detallada en [3]: en essència, aquesta operació itera cap a un conjunt de fases òptim pel qual el factor de cresta del senyal multisinus prèviament especificat és mínim. El resultat d'aquesta operació és un vector d'amplituds complexes de la multisinus sobre les quals es computen una sèrie d'operacions que permeten obtenir el vector final de valors d'entrada.

Aquesta entrada és la que es farà servir per a calcular-ne la sortida. A partir d'aquests senyals d'entrada i sortida ja es podrà obtenir el model corresponent.

A continuació es presenta el senyal multisinus generat per a fer la identificació del model.

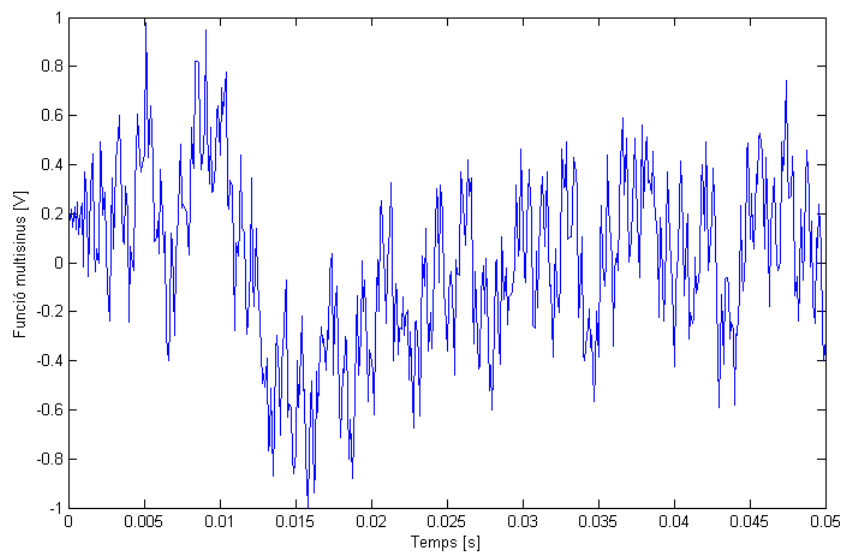


Figura 2.1: Senyal multisinusoïdal

## 2.3 Mètodes d'ajust

Un cop es disposa del fitxer que conté els valors de les variables enregistrades, la identificació del model del motor comença pròpiament. Aquesta identificació es durà a terme seguint dos mètodes diferents: d'una banda, es farà servir la *identification toolbox*, de caràcter més visual; de l'altra, es segueix el mètode *output-error*, àmpliament descrit en [13]. Aquesta forma de procedir té una doble finalitat: primerament, es podran comparar directament els resultats obtinguts per a cada mètode, de forma que es podrà saber si aquests són coherents o no; a més, és una forma de conèixer diverses alternatives a l'hora d'ajustar models.

### 2.3.1 *Identification Toolbox*

La *identification toolbox* és una eina del MATLAB que permet obtenir el model matemàtic que millor representa un conjunt de dades de manera intuïtiva. Aquesta operació es du a terme en tres etapes:

**Import data** llegeix els paràmetres prèviament enregistrats al fitxer i bolcats al *MATLAB*.

En el present cas, es selecciona la opció *Time domain data*, ja que les dades provenen d'un experiment real.

**Preprocess** permet fer diverses correccions sobre les dades abans de fer-les servir per a estimar el model. Entre d'altres, es pot eliminar el valor mig, tornar a mostrejar o seleccionar el rang de dades que es vol estudiar. En aquest treball no es considera cap procés intermedi.

**Estimate** computa pròpiament la identificació del model. Es poden realitzar diversos tipus d'estimació, segons la forma en què es desitgi obtenir el model: en forma d'espai d'estats, com a funció de transferència, com a model polinòmic, etc. En aquest cas, s'opta per obtenir un *process model*; és a dir, una funció de transferència en temps continu que descriu la dinàmica d'un sistema lineal.

Aquesta funció de transferència es caracteritza per mitja d'una sèrie de paràmetres. Segons es triïn uns paràmetres o altres, així com la seva multiplicitat, el model obtingut serà diferent en cada cas. Dits paràmetres són:

$K$  guany estàtic

$T_{pk}$   $k$  constants de temps, per al cas de pols reals, que són els que s'estudien.

$T_{zj}$   $j$  constants de temps, per al cas de zeros reals.

$T_d$  retard pur, que representa el temps que la sortida triga a respondre envers l'entrada.

$s_l$   $l$  integradors purs.

Com s'ha vist en l'apartat 2.1, l'estructura del motor d'*Ingenia* és:

$$G(s) = \frac{K}{s(\tau s + 1)} \quad (2.9)$$

en què  $G(s)$  relaciona el corrent que circula pel motor - directament relacionat amb el parell proporcionat a l'entrada - i la posició angular a la sortida.

Per tant, d'acord amb el que s'acaba d'exposar, cal triar:

$k = 1$  un pol real

$j = 0$  cap zero

$l = 1$  un integrador pur

És clar que  $k + l = 2$ . Ara bé, és essencial incidir en el fet que el model que s'obtingria en seleccionar  $k = 2$  i  $l = 0$  no coincidiria exactament amb el model obtingut en fixar  $k = l = 1$ , si bé en tots dos casos el sistema té dos pols. En la segona configuració es força que un dels

dos pols sigui un integrador pur, i per això es selecciona aquesta alternativa; de no fer-ho així, el model obtingut no seria vàlid.

Si bé aquest procediment és correcte des del punt de vista teòric, es detectaran comportaments anòmals en els resultats. Es detectarà, a còpia de fer molts experiments, la necessitat de retocar el codi per tractar prèviament les dades abans de fer-ne l'ajust. Aquest tractament, que es detalla en l'apartat 2.3, consistirà a eliminar l'efecte de l'integrador sobre les mesures obtingudes a la sortida.

Finalment, adoptant la notació que empra la *toolbox*, l'estructura sobre la que s'ajusten les dades és:

$$G(s) = \frac{K_p}{T_{z1}s + 1} \quad (2.10)$$

Per tant, si es compara el model final del motor d'*Ingenia* descrit en l'equació (Eq. 2.10) amb el model del motor del Laboratori de l'equació (Eq. 2.8), s'observa que s'està ajustant la mateixa estructura, tot i que no ho semblés així en comparar les equacions (Eq. 2.7) i (Eq. 2.8). L'única diferència radica en què cal tractar prèviament les dades del motor d'*Ingenia* per a eliminar l'integrador, ja que aquest no disposa de sensor de velocitat, a diferència del motor del Laboratori, que disposa d'un tacòmetre.

### 2.3.2 Output-Error

*Output-error* - OE d'ara en endavant - designa un tipus d'estructura de model lineal que s'obté en estimar els paràmetres d'un model a partir de les mesures fetes a l'entrada i a la sortida. Tal i com s'explica en [12], hi ha tres tipus de models en el camp de la identificació de sistemes:

**Black-box Model** engloba aquells casos en què no es coneix el sistema des d'un punt de vista físic; això és, el conjunt d'equacions que el regeixen. La filosofia, doncs, consisteix a provar un conjunt d'estructures disponibles per a la identificació: a través de l'assaig i error, l'usuari haurà de determinar l'alternativa que millor s'adapti a les seves dades.

**Grey-box Model** es diferencia del cas anterior en què aquí l'usuari coneix part de la dinàmica o dels paràmetres del sistema. Per tant, s'especifica la part coneguda del model i s'estima la part desconeguda.

**User-defined Model** es fa servir quan els casos anteriors no són capaços de representar el sistema físic que es desitja estimar, especialment si aquest conté no-linealitats.



Si bé OE s'engloba dins de la categoria *Black-box Model* en tant que permet estimar el model sense conèixer les equacions físiques que governen el sistema, també es pot emprar-lo quan es coneix l'estructura del model, com succeeix en el present cas d'estudi. En l'apartat 2.1 es va deduir, a partir d'una versió simplificada de les equacions, quina havia de ser l'estructura final del model. Per tant, en usar l'estructura OE no caldrà seguir aquest procés de prova i error típic dels *Black-box Model*, sinó que es seleccionarà directament l'estructura desitjada.

Tal i com es descriu en la secció 2-23 de [10], l'estructura general d'un model lineal entrada-sortida de tipus *MISO - Multiple-Inputs Single-Output* - és

$$A(q)y(t) = \sum_{i=1}^{nu} \frac{B_i(q)}{F_i(q)} u_i(t - nk_i) + \frac{C(q)}{D(q)} e(t) \quad (2.11)$$

en què  $u_i$  denota l'entrada i-èsima;  $y(t)$ , la sortida;  $e(t)$ , la pertorbació; i  $A, B_i, C, D$  i  $F_i$ , polinomis expressats en l'operador de decalatge ( $z$  o  $q$ ). Aquests polinomis permeten expressar de forma compacta les equacions diferencials del sistema.

L'equació (Eq. 2.11) es pot expressar en forma de diagrama de blocs. Es mostra el diagrama per al cas d'estudi, que no és de tipus *MISO* sinó *SISO*; és a dir, *Single-Input Single-Output*.

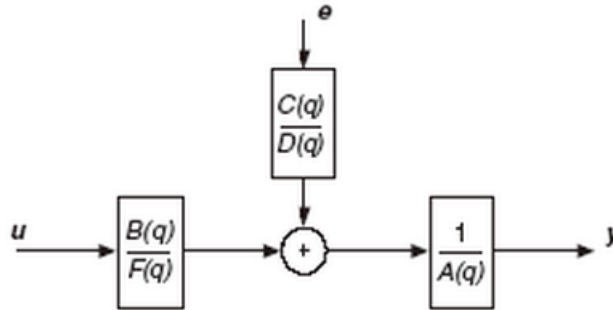


Figura 2.2: Estructura general del model lineal

La figura 2.2 contempla l'estructura més general possible per a un sistema lineal de tipus *SISO*. Ara bé, tot sovint s'empren casos especials que s'adapten millor a les dades que es volen ajustar, i alguns dels més utilitzats són: AR (*AutoRegressive*), ARX (*AutoRegressive with eXogenous inputs*), ARMAX (*AutoRegressive-Moving-Average with eXogenous inputs*), BJ (*Box-Jenkins*) i OE (*Output-Error*).

L'estructura OE per a un sistema *SISO* és descrita a la secció 3-12 de [10] com:

$$y(t) = \frac{B(q)}{F(q)}u(t - nk) + e(t) \quad (2.12)$$

I el seu corresponent diagrama de blocs és:

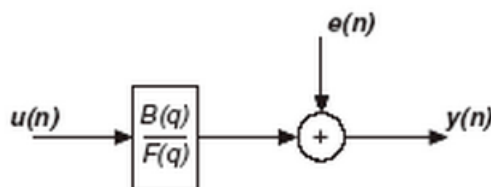


Figura 2.3: Estructura del model *OE*

La tria de l'estructura OE per a ajustar les dades obeeix a diversos motius:

1. En general, acostuma a ser el mètode que millors resultats proporciona, tot i que l'èxit en la seva aplicació depèn de les dades experimentals de partida. Es pot començar provant l'estructura OE, i si s'escau es canvia d'estructura.
2. Proporciona una bona descripció de la part dinàmica del sistema, encara que no es modela la pertorbació (secció 3-79 de [10]).
3. Només utilitza valors passats de l'entrada per a predir els valors futurs de la sortida (secció 4-137 de [10]).

Un cop s'opta per aquesta estructura, el següent pas és comprendre com funciona la identificació sobre aquesta estructura. En essència, aquest mètode, àmpliament descrit en els capítols 7 i 8 de [13], resol un problema d'optimització, obtenint el valor dels paràmetres del model que es volien estimar i que minimitzen el valor d'una funció de cost. El sentit de l'optimització sobre aquesta funció força que els valors que s'estimen per als paràmetres siguin tals que la diferència quadràtica entre la sortida experimental i la sortida estimada -calculada a partir d'aquests paràmetres estimats- sigui mínima.

Aquest és el procediment intern que segueix el *MATLAB*, però a efectes pràctics interessa la solució del problema; és a dir, els valors dels paràmetres i el grau d'ajust o *fitting* entre les dades mesurades i el model obtingut a partir de l'estimació dels paràmetres. Cal fer servir, tal i com es descriu en [11], la comana *oe*. Sigui

$$sys = oe(data, [nb \ nf \ nk], Name, Value)$$

l'estructura de dita comana, si es proporcionen els arguments *data*, [nb nf nk] i la parella (*Name*, *Value*), es computa la operació *oe* i s'obté *sym*. A continuació es descriuen tots aquests elements:

***data*** conté els valors experimentals d'entrada i de sortida de la planta a partir de la qual es volen estimar els paràmetres del model. Segons si es desitja fer l'ajust en el domini del temps o de la freqüència, l'estructura de *data* és lleugerament diferent en un o altre cas. S'ajustarà un model en cada domini amb l'objectiu de veure quina versió proporciona millors resultats.

**nb nf nk** són els ordres del model que es vol estimar. Es recorda que

$$y(t) = \frac{B(q)}{F(q)}u(t - nk) + e(t) \quad (2.13)$$

és la representació del sistema - vist en l'equació (Eq. 2.12) - per a un sistema de tipus *SISO*. Així:

*nb* és l'ordre del polinomi  $B(q) + 1$ .

*nf* ordre del polinomi  $F(q)$ . Els pols de  $F(q)$  són exclusius per a la dinàmica provinent de l'entrada  $u(t)$  (secció 2-24 de [10]).

*nk* expressa el retard del sistema. A causa del mostreig introduït en passar de temps continu a temps discret, *nk* és igual a 1 (secció 2-21 de [10]).

***Name*, *Value*** és la parella d'arguments a través de la qual es poden afegir opcions addicionals. S'ha comentat anteriorment que els models ajustats s'obtindran en temps discret, per la qual cosa cal mostrejar les dades. Així doncs, s'especifica aquí el període de mostreig  $T_s$ .

***sys*** recull el resultat de computar la operació d'ajust sobre la informació anterior. Per a cada acció de còmput s'obté el model ajustat i un *fitting* entre les dades emprades per a l'ajust i el model ajustat.

Es va deduir en l'apartat 2.1 quines havien de ser les estructures dels models ajustats per a un i altre motor. Ara bé, en l'equació (Eq. 2.10) es posava de manifest que el model ajustat per

als dos motors seria el mateix, requerint un tractament previ de les dades del motor d'*Ingenia*. Segons tot això, és clar que  $nb = 1$ ,  $nf = 1$  i  $nk = 1$ .

## 2.4 Resultats de la identificació

Un cop detallats els dos mètodes que s'empraran per a fer la identificació del model del motor, cal obtenir les dades sobre les quals es provaran els ajustos. Tal i com s'ha comentat a l'inici de l'apartat 2.2, el fet de fer servir dos mètodes diferents permetrà comprovar si el model obtingut és consistent. Cal, doncs, fer els experiments per a generar els jocs de dades, desenvolupar els codis per fer l'ajust via OE - es poden trobar a l'annex C.1.2 - i aplicar les dues tècniques d'ajust.

Sobre cadascun dels dos motors que constitueixen l'objecte d'estudi del treball - motor d'*Ingenia* i motor del Laboratori de Control - es realitzen una sèrie d'ajustos. Aquests ajustos es realitzaran amb dades provinents directament de l'experimentació amb els motors. Abans de fer això, cal assegurar-se que la metodologia d'ajust sigui la correcta: per tant, es du a terme una identificació prèvia en simulació sobre el motor del Laboratori, del qual se'n pot conèixer l'expressió de la seva funció de transferència a partir d'un experiment molt senzill en llaç obert. L'objectiu d'aquesta idea és clara: es desitja veure si la reconstrucció del model a partir de l'entrada i la sortida coincideix amb el model prèviament conegut.

Per altra banda, també es vol conèixer l'efecte que té executar l'ajust en llaç obert o en llaç tancat, i saber en quin cas s'obtenen millors resultats.

Per tal de facilitar al lector l'assimilació de tota aquesta informació, es detalla a continuació l'ordre i les proves d'identificació que s'han realitzat:

1. Simulació: identificació en llaç obert sobre el model del motor del Laboratori.

2. Experiments:

- (a) Motor del Laboratori: identificació en llaç obert.

- (b) Motor d'*Ingenia*:

- Identificació en llaç obert.
- Identificació en llaç tancat, traient l'integrador.
- Identificació en llaç tancat, sense treure l'integrador.

La obtenció d'aquests resultats no ha estat senzilla ni directa, sinó que va ser necessari un procés gens menyspreable de depuració del mètode; sobre això ja se n'ha parlat amb anterioritat, i ara s'amplia degudament. Es va indicar que els ordres previstos per al OE eren  $nb = 1$ ,  $nf = 1$  i  $nk = 1$  per als dos motors, relacionant així el corrent i la velocitat angular. En no disposar el motor d'*Ingenia* d'un sensor de velocitat angular, s'havia d'emprar la posició i no la velocitat per a construir el model, fet que comportava l'aparició d'un integrador pur al model i que complicava l'obtenció d'un bon model.

No obstant aquest fet, si es sap que la planta conté un integrador pur, tal i com es comenta a l'apartat 7.2.2 de [8], es pot aprofitar aquesta informació per reduir la complexitat del model i, sobretot, incrementar la precisió de la identificació: en el present cas d'estudi, aquest increment de precisió és molt important, com es mostrarà més endavant.

Els autors de [8] proposen dues maneres equivalents d'eliminar l'efecte de l'integrador; a continuació es detalla l'alternativa triada.

Sigui  $u(t)$  l'entrada i  $y(t)$  la sortida, es pot aproximar  $y(t)$  per la seva derivada a través de l'aproximació en diferències cap endarrere:

$$y'(t) = \frac{y(t) - y(t-1)}{\Delta t} \quad (2.14)$$

essent  $\Delta t$  la diferència de temps entre els valors  $y(t)$  i  $y(t-1)$ ; és a dir, el període de mostreig  $T_s$ . Per tant, s'identifica el model a partir dels senyals  $u(t)$  i  $y'(t)$ .

A efectes pràctics, la conseqüència d'aquesta aproximació és que aquesta derivada cancel·la l'integrador pur que es contemplava en l'equació (Eq. 2.6). Per tant, el model a identificar pren l'estructura

$$G(s) = \frac{K}{\tau s + 1} \quad (2.15)$$

com ja s'havia vist en l'equació (Eq. 2.10).

En eliminar l'integrador s'està derivant la sortida - la posició angular - i per tant el model de l'equació (Eq. 2.15) relaciona el corrent i la posició angular.

Inicialment no es coneixia aquest fenomen, la qual cosa va propiciar l'obtenció de models ajustats realment dolents. Per tal que el lector se'n pugui fer una millor idea, en l'apartat de resultats experimentals sobre el motor d'*Ingenia* en llaç tancat es presentaran els ajustos obtinguts d'ambdues maneres; és a dir, eliminant i sense eliminar l'efecte de l'integrador sobre la sortida del sistema. Per al darrer cas es presentaran, a més, els pols de la funció de transferència

ajustada, amb l'objectiu de verificar si s'està identificant bé l'integrador -  $z = 1$  en temps discret. Només es farà per als ajustos amb el mètode OE, ja que l'ajust mitjançant *toolbox* permet especificar-ho prèviament.

A continuació es presenten els resultats de tots els casos que s'han llistat anteriorment. Per a cada cas s'inclouen les següents figures i informacions:

**Diagrama de Bode de l'ajust amb la *toolbox ident*.** S'hi representen:

- Funció de transferència del motor en temps continu - obtinguda amb la *toolbox ident*, de color blau.
- Funció de transferència del motor en temps discret - obtinguda per discretització de la de temps continu, de color vermell.
- Conjunt de punts emprats per a fer l'ajust.

Es presenten la funció de transferència obtinguda en temps continu a través de la *toolbox ident*, la funció de transferència en temps discret obtinguda a partir de la de temps continu i el *fitting* entre dita funció i les dades d'ajust.

**Diagrama de Bode de l'ajust *oe temporal*.** S'hi representen:

- Funció de transferència del motor - obtinguda amb la *toolbox ident* - en temps discret, de color verd. Serveix per a comparar visualment els dos mètodes.
- Ajust *oe* en el domini temporal, en blau.
- Conjunt de punts emprats per a fer l'ajust.

Es presenten la funció de transferència en temps discret obtinguda a partir del *oe temporal*, i el *fitting* entre dita funció i les dades d'ajust.

**Diagrama de Bode de l'ajust *oe freqüencial*.** S'hi representen:

- Funció de transferència del motor - obtinguda amb la *toolbox ident* - en temps discret, de color verd. Això només per als resultats experimentals. Serveix per a comparar visualment els dos mètodes.
- Ajust *oe* en el domini freqüencial, en blau.
- Conjunt de punts emprats per a fer l'ajust.

Es presenten la funció de transferència en temps discret obtinguda a partir de la freqüencial, i el *fitting* entre dita funció i les dades d'ajust.

### 2.4.1 Verificació del mètode en simulació

Com s'ha comentat abans, es prova el mètode d'ajust fent servir l'eina *Simulink* per a substituir el motor real per una representació física d'aquest. El model que representa el motor es pot obtenir fàcilment fent un experiment en llaç obert - consultar [4] per a més informació - amb una consigna graó sobre el motor: com que es tracta d'un sistema de primer ordre, es poden identificar el guany i la constant de temps del sistema per inspecció directa. Després, només cal fer-ne la discretització per obtenir-ne l'expressió en temps discret. Les equacions són:

$$G_{motor}(s) = \frac{1.02}{0.3s + 1}$$

$$G_{motor}(z) = \frac{0.03344}{z - 0.9672}$$

Per a fer la discretització s'empra un període de mostreig de 0.01 segons, adequat per a aquest tipus de motor. El motor d'*Ingenia*, però, és més ràpid que el del Laboratori, i per això es farà servir un període de 0.001 segons.

Els resultats de la validació del mètode d'identificació són:

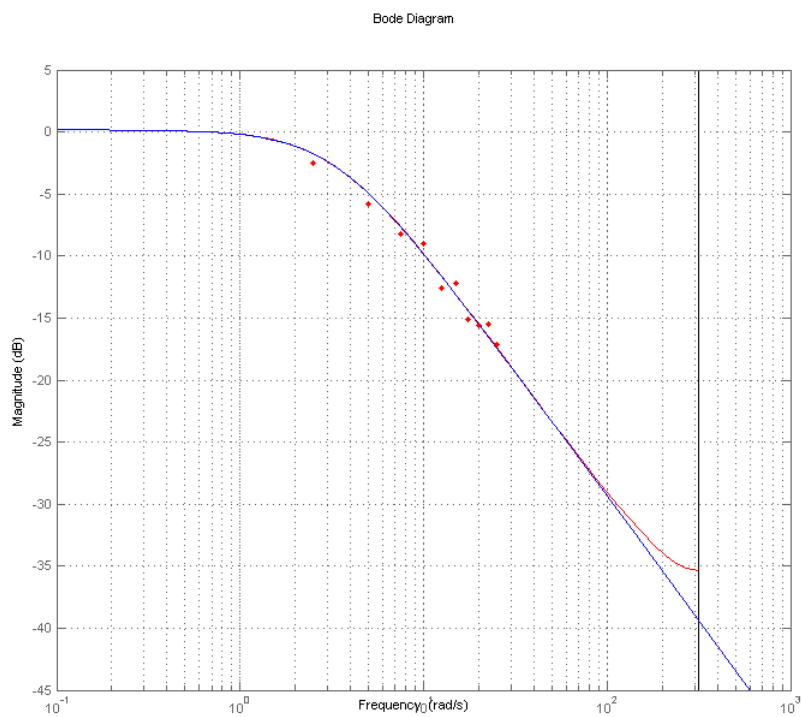


Figura 2.4: Diagrama de Bode de l'ajust amb *toolbox ident*

$$G_{\text{toolbox}}(s) = \frac{1.0194}{0.29551s + 1}$$

$$G_{\text{toolbox}}(z) = \frac{0.03392}{z - 0.9667}$$

*Fitting* : 95.84%



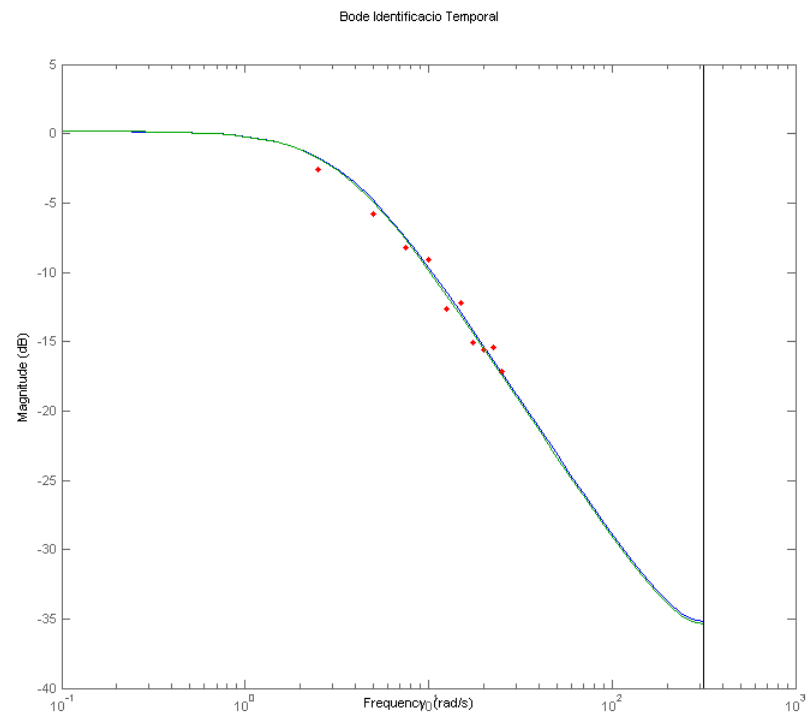


Figura 2.5: Diagrama de Bode de l'ajust OE temporal

$$OE_{temporal}(z) = \frac{0.0341}{z - 0.9666}$$

*Fitting* : 95.82%

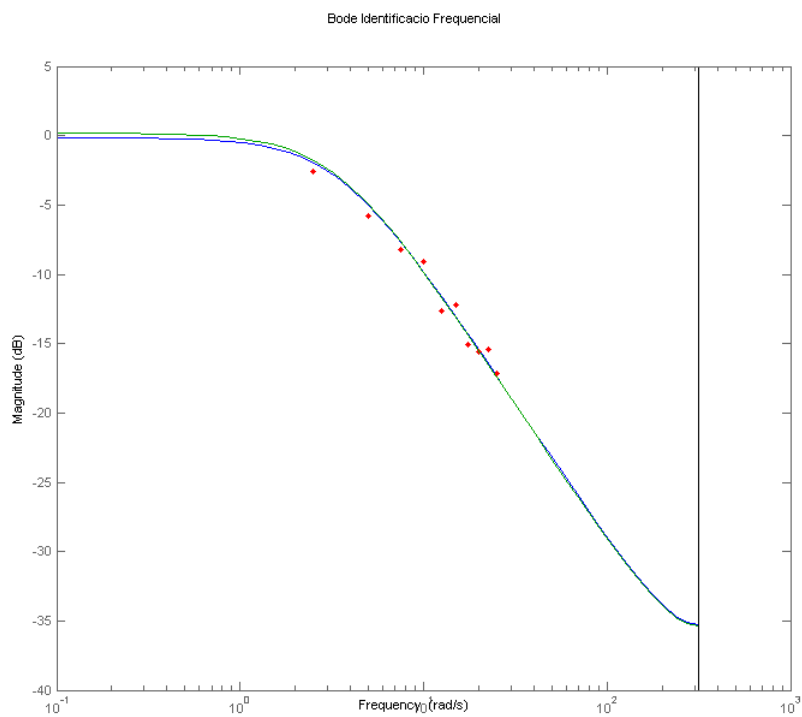


Figura 2.6: Diagrama de Bode de l'ajust OE freqüencial

$$OE_{freqüencial}(z) = \frac{0.03372}{z - 0.9657}$$

*Fitting* : 90.02%

### 2.4.2 Resultats experimentals sobre el motor del Laboratori (llaç obert)

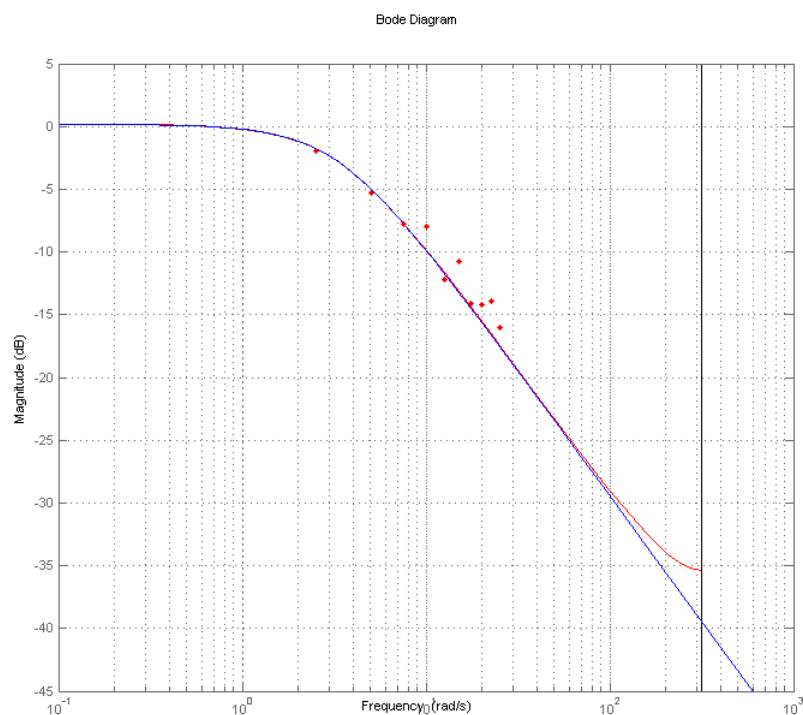


Figura 2.7: Diagrama de Bode de l'ajust amb *toolbox ident*

$$G_{toolbox}(s) = \frac{1.0126}{0.27322s + 1}$$

$$G_{toolbox}(z) = \frac{0.03639}{z - 0.9641}$$

*Fitting* : 94.21%

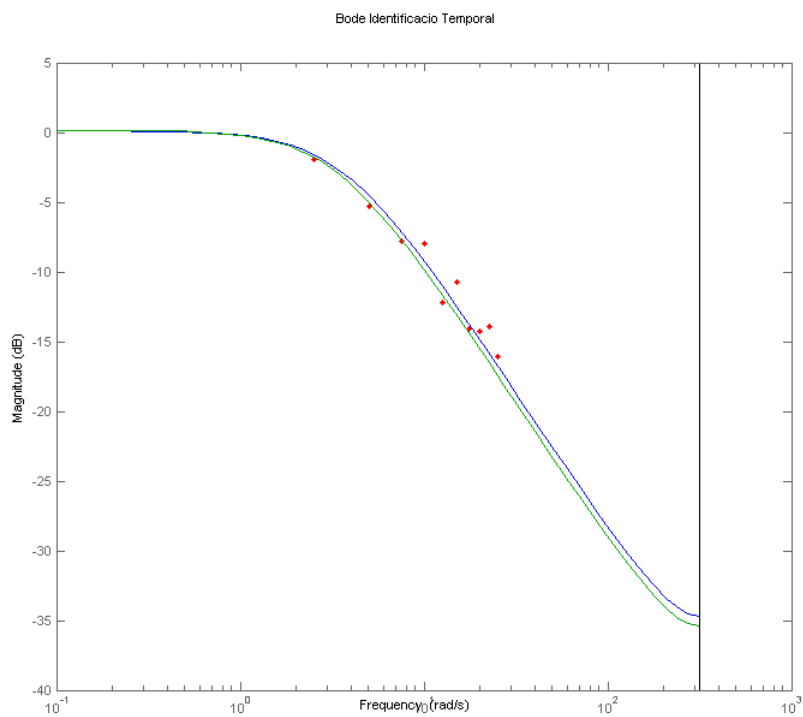


Figura 2.8: Diagrama de Bode de l'ajust OE temporal

$$OE_{temporal}(z) = \frac{0.03622}{z - 0.9642}$$

*Fitting* : 94.19%

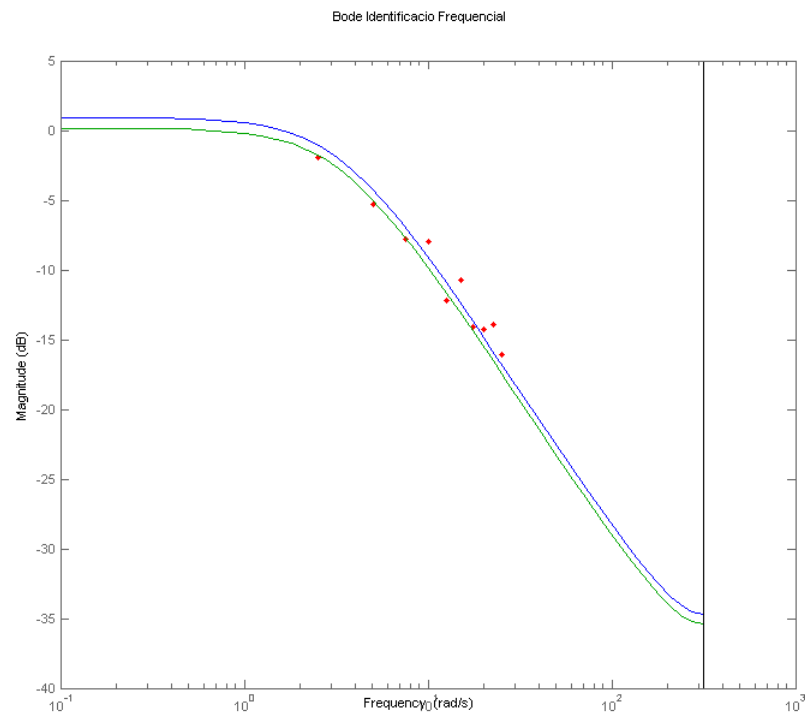


Figura 2.9: Diagrama de Bode de l'ajust OE freqüencial

$$OE_{freqüencial}(z) = \frac{0.03634}{z - 0.9673}$$

*Fitting* : 85.44%

### 2.4.3 Resultats experimentals sobre el motor d'*Ingenia*

En llaç obert, traient l'integrador

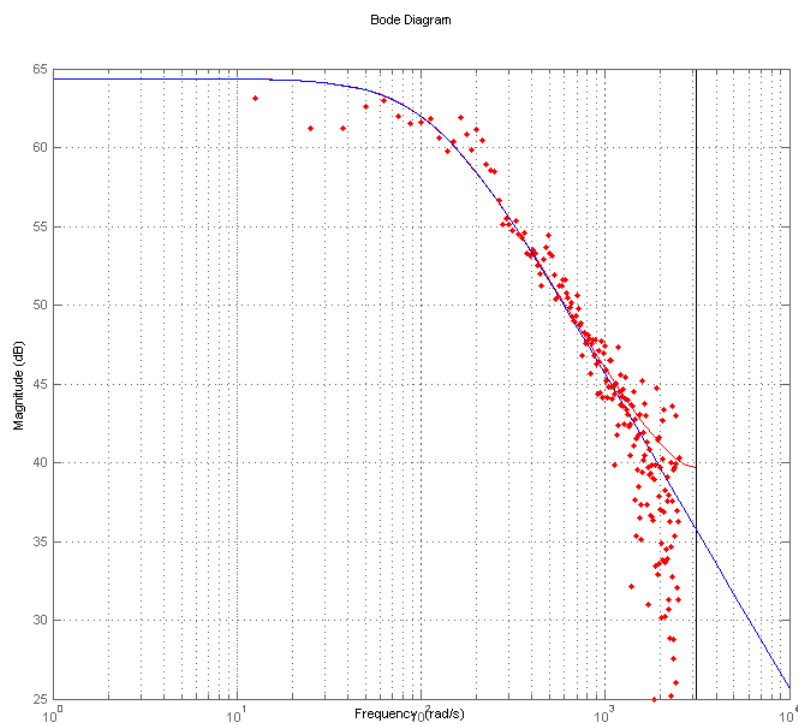


Figura 2.10: Diagrama de Bode de l'ajust amb *toolbox ident*

$$G_{\text{toolbox}}(s) = \frac{1656}{0.00859s + 1}$$

$$G_{\text{toolbox}}(z) = \frac{182}{z - 0.8901}$$

*Fitting* : 68.32%

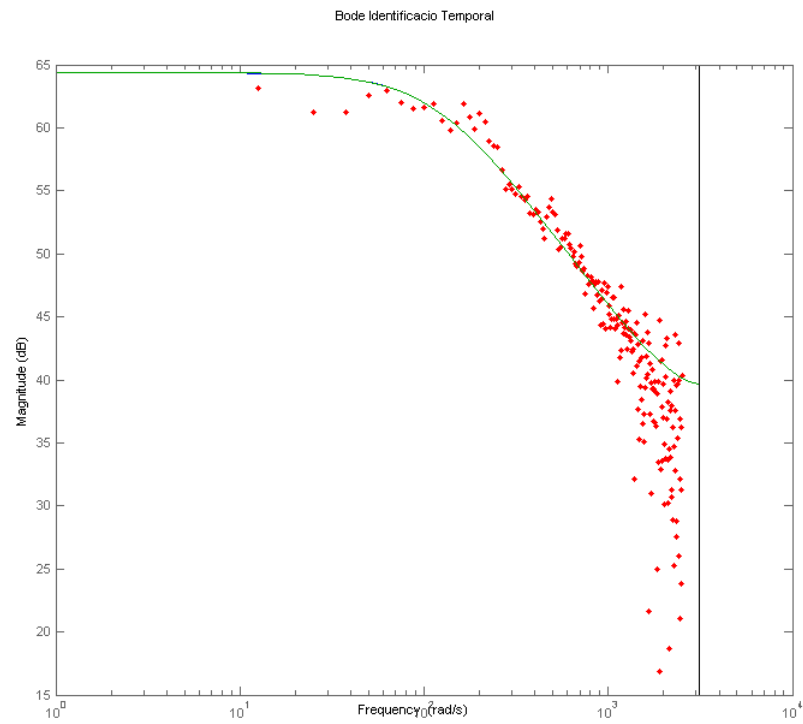


Figura 2.11: Diagrama de Bode de l'ajust OE temporal

$$OE_{temporal}(z) = \frac{182}{z - 0.89}$$

*Fitting* : 68.3%

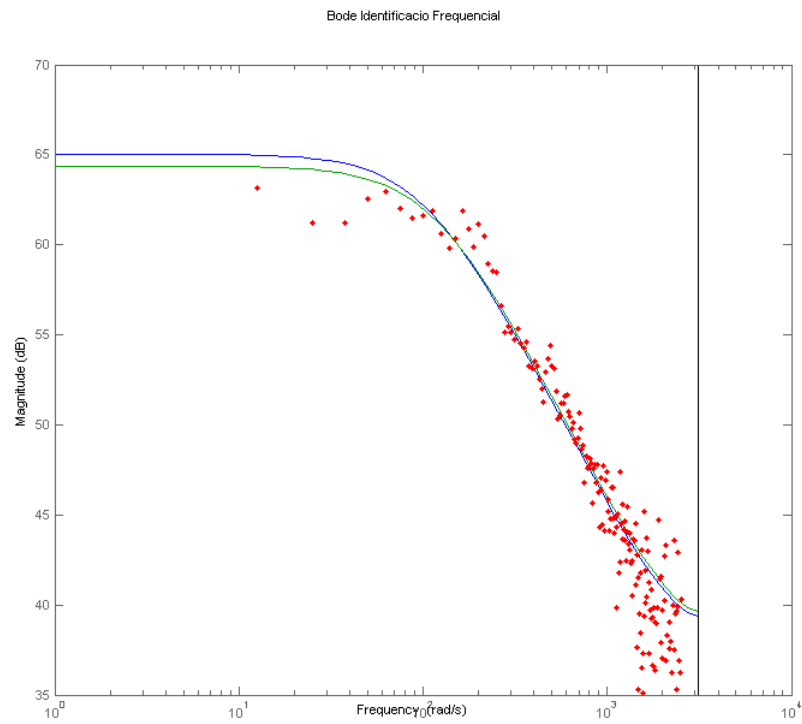


Figura 2.12: Diagrama de Bode de l'ajust OE freqüencial

$$OE_{freqüencial}(z) = \frac{176.9}{z - 0.9009}$$

*Fitting* : 63.15%



Enllaç tancat, traient l'integrador

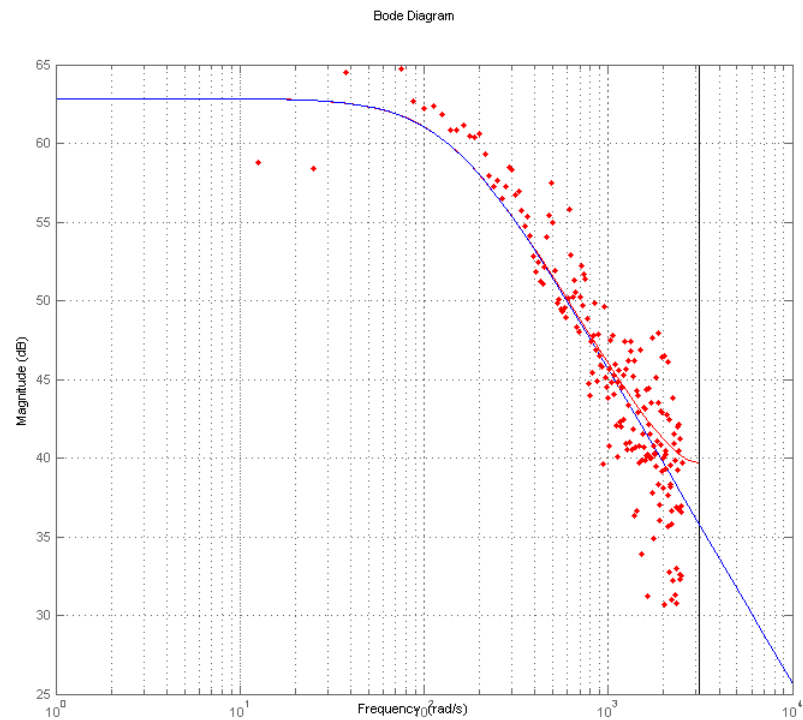


Figura 2.13: Diagrama de Bode de l'ajust amb *toolbox ident*

$$G_{\text{toolbox}}(s) = \frac{1390}{0.00719s + 1}$$

$$G_{\text{toolbox}}(z) = \frac{180.5}{z - 0.8702}$$

*Fitting* : 54.16%

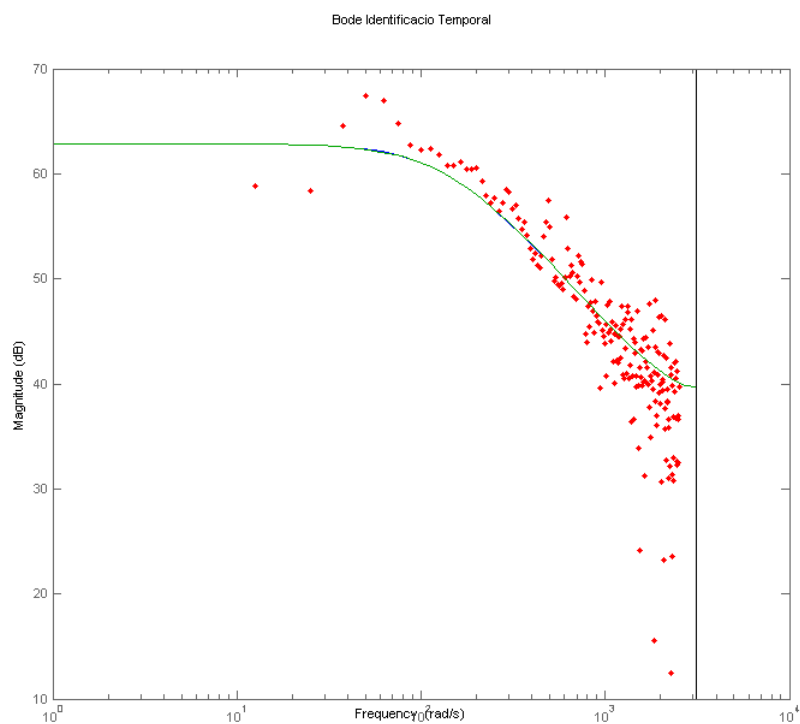


Figura 2.14: Diagrama de Bode de l'ajust OE temporal

$$OE_{temporal}(z) = \frac{180.4}{z - 0.8704}$$

*Fitting* : 54.15%

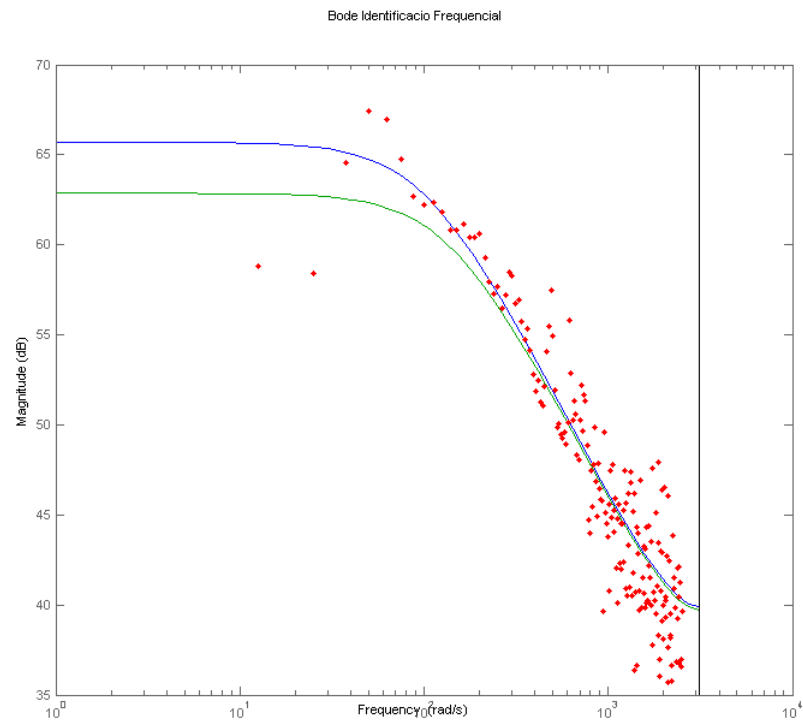


Figura 2.15: Diagrama de Bode de l'ajust OE freqüencial

$$OE_{freqüencial}(z) = \frac{178.6}{z - 0.9026}$$

*Fitting* : 71.85%

En llaç tancat, sense treure l'integrador

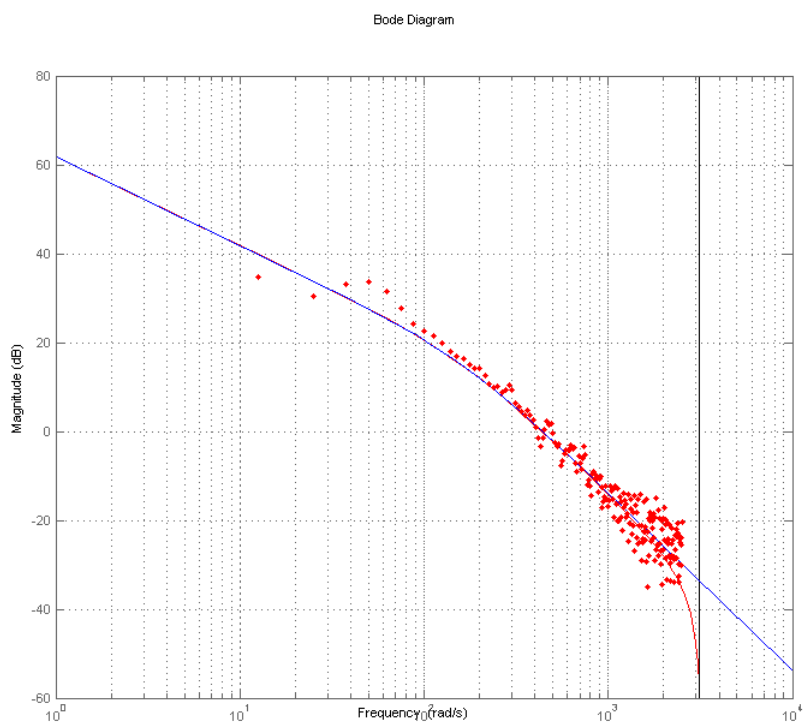


Figura 2.16: Diagrama de Bode de l'ajust amb *toolbox ident*

$$G_{\text{toolbox}}(s) = \frac{1232.8}{0.00603s^2 + s}$$

$$G_{\text{toolbox}}(z) = \frac{0.09678s + 0.09158}{z^2 - 1.847z + 0.8472}$$

*Fitting* : 44.19%

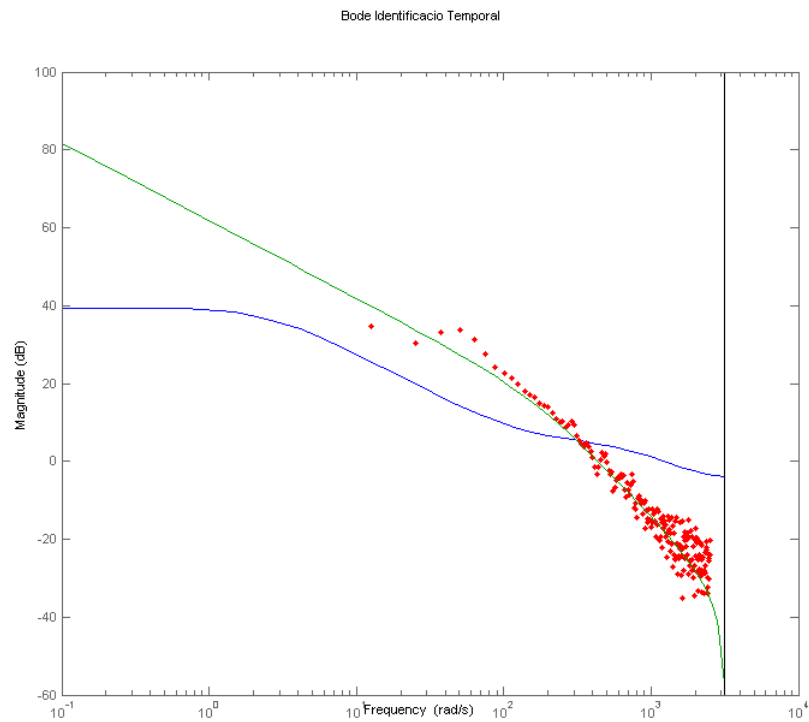


Figura 2.17: Diagrama de Bode de l'ajust OE temporal

$$OE_{temporal}(z) = \frac{-1.014z + 0.8914}{z^2 - 1.49z + 0.491}$$

$$pols(OE_{temporal}(z)) : z_1 = 0.9974; z_2 = 0.4922$$

$$Fitting : -29.33\%$$

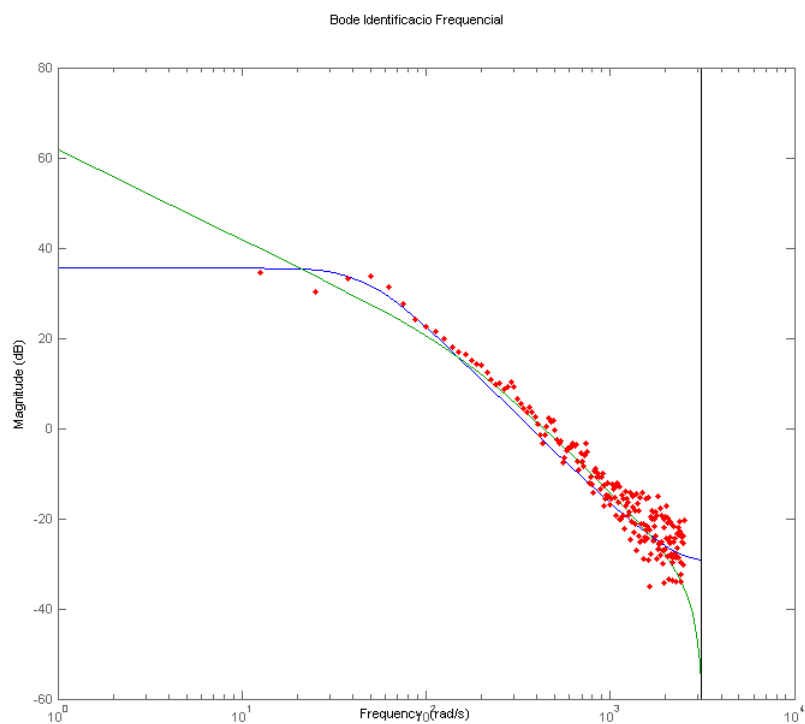


Figura 2.18: Diagrama de Bode de l'ajust OE freqüencial

$$OE_{freqüencial}(z) = \frac{0.1339z - 0.00175}{z^2 - 1.929z + 0.9313}$$

$$pols(OE_{freqüencial}(z)) : z_{1,2} = 0.9645 \pm j0.0306$$

*Fitting* : 70.1%

## 2.4.4 Resum dels resultats

MOTOR DEL LABORATORI			
Model conegut	$G(s) = \frac{1.02}{0.3s + 1}$	$G(z) = \frac{0.03344}{z - 0.9672}$	
	<b>Toolbox ident</b>	<b>OE temporal</b>	<b>OE freqüencial</b>
Verificació en simulació	$G(s) = \frac{1.0194}{0.29551s + 1}$		
	$G(z) = \frac{0.03392}{z - 0.9667}$ <i>Fitting</i> : 95.84%	$OE(z) = \frac{0.0341}{z - 0.9666}$ <i>Fitting</i> : 95.82%	$OE(z) = \frac{0.03372}{z - 0.9657}$ <i>Fitting</i> : 90.02%
Experiment en llac obert	$G(s) = \frac{1.0126}{0.27322s + 1}$		
	$G(z) = \frac{0.03639}{z - 0.9641}$ <i>Fitting</i> : 94.21%	$OE(z) = \frac{0.03622}{z - 0.9642}$ <i>Fitting</i> : 94.19%	$OE(z) = \frac{0.03634}{z - 0.9673}$

Taula 2.1: Models identificats per al motor del Laboratori

MOTOR D'INGENIA	<i>Toolbox ident</i>	OE temporal	OE freqüencial
Llaç obert (sense integrador)	$G(s) = \frac{1656}{0.00859s + 1}$ $G(z) = \frac{182}{z - 0.8901}$ <i>Fitting</i> : 68.32%	$OE(z) = \frac{182}{z - 0.89}$ <i>Fitting</i> : 68.3%	$OE(z) = \frac{176.9}{z - 0.9009}$ <i>Fitting</i> : 63.15%
Llaç tancat (sense integrador)	$G(s) = \frac{1390}{0.00719s + 1}$ $G(z) = \frac{180.5}{z - 0.8702}$ <i>Fitting</i> : 54.16%	$OE(z) = \frac{180.4}{z - 0.8704}$ <i>Fitting</i> : 54.15%	$OE(z) = \frac{178.6}{z - 0.9026}$ <i>Fitting</i> : 71.85%
Llaç tancat (amb integrador)	$G(s) = \frac{1232.8}{0.00603s^2 + s}$ $G(z) = \frac{0.09678s + 0.09158}{z^2 - 1.847z + 0.8472}$ <i>Fitting</i> : 44.19%	$OE(z) = \frac{-1.014z + 0.8914}{z^2 - 1.49z + 0.491}$ <i>Fitting</i> : -29.33%	$OE(z) = \frac{0.1339z - 0.00175}{z^2 - 1.929z + 0.9313}$ <i>Fitting</i> : 70.1%

Taula 2.2: Models identificats per al motor d'Ingenia



## 2.5 Conclusions sobre la identificació

Aquest primer capítol conclou amb l'anàlisi final dels diversos resultats presentats en l'apartat anterior.

En primer lloc, resulta interessant establir una comparació entre el comportament d'un i altre motor. Es pot observar com el motor del Laboratori presenta *fittings* més alts que no pas el motor d'*Ingenia*. És cert que en el motor del Laboratori s'han generat menys punts per a fer la identificació, però això es deu al fet que interessa identificar sobretot aquest rang de freqüències, que correspon a l'interval assolible per a dit motor. Pel que fa al motor d'*Ingenia*, les dades experimentals per a ajustar-ne els paràmetres del model es van obtenir directament de l'empresa. Cal observar que, a les altes freqüències, els punts deixen de seguir el model: això es deu a que dites mesures s'han efectuat cap al final del recorregut - es recorda que es tracta d'un motor lineal amb un recorregut limitat -, i possiblement estiguin afectades per aquest fet.

En relació amb aquest primer punt, cal tenir en compte que existeix una diferència molt important entre els dos motors i que pot ser la causant que s'hagin obtingut models amb millor *fitting* per al motor del Laboratori que no pas per al motor d'*Ingenia*: el primer disposa d'un sensor de velocitat, i el segon, no. En tant que no apareix cap integrador, que és l'element que distorsiona l'ajust, si es fa servir la velocitat angular com a sortida del sistema, es posa de manifest la necessitat de treure l'efecte de l'integrador sobre les dades del motor d'*Ingenia*. Només cal comparar els resultats en un i altre cas - Figura 2.13 a Figura 2.18 - per veure'n la seva importància.

Per altra banda, tant les funcions de transferència com les gràfiques dels resultats experimentals permeten comprovar com la *toolbox ident* i el mètode *output-error* proporcionen models molt semblants. Només cal comparar les corbes verda - model obtingut amb *toolbox ident* - i blava - model obtingut amb *output-error* - per a verificar aquesta afirmació. També es pot observar com el mètode OE freqüencial, a diferència del temporal, presenta lleugeres diferències pel que fa al guany estàtic, que disminueixen a mesura que la freqüència augmenta.

Cal no oblidar que l'estudi efectuat en aquest apartat ha de servir per a desenvolupar controladors adequats, que permetin que el motor es comporti tal i com dicten les especificacions desitjades. S'ha vist com, en general, eren els ajustos OE en el domini temporal els que proporcionaven millors resultats, i per tant sembla lògic dissenyar les estructures de control sobre els models obtinguts en el domini OE temporal.



## Capítol 3

# Control

El present capítol conforma la segona part del treball, dedicada al disseny de controladors per a les plantes prèviament identificades al Capítol 2. L'objectiu principal d'aquest apartat és, doncs, estudiar i provar diverses estructures de control sobre els motors per tal que aquests es comportin d'acord amb les especificacions desitjades.

El capítol està estructurat de la següent manera: primerament es detallen les especificacions de funcionament per a aquest motor, així com les estructures de control que es consideren per a la implementació. A continuació es dedueixen les relacions entre els paràmetres del controlador i les especificacions inicials a través de l'assignació de pols, parant especial atenció al problema de la saturació. Finalment es presenten tots els resultats obtinguts en aplicar els controladors, tant en simulació com experimentalment, i s'analitzen adequadament tot comentant les dificultats trobades, obrint així una porta a una futura via d'estudi.

### 3.1 Definició de les especificacions

*Ingenia* dissenya els controladors en base a les especificacions dels clients, que en el context del *motion control* acostumen a ser referides al *bandwidth* o ample de banda. S'entén aquest ample de banda - [14] - com el rang de freqüències que el senyal d'entrada pot assumir mentre es passi de l'entrada a la sortida sense una atenuació de la magnitud excessiva. En tant que la majoria de sistemes - i aquest no n'és una excepció - presenten una naturalesa de tipus passa-baixos - les baixes freqüències *passen* sense patir pràcticament atenuació -, l'ample de banda es defineix com l'interval de freqüències que parteixen de 0 - corrent continu - i arriben a la màxima freqüència  $\omega_{BW}$ , també coneguda com a *cutoff frequency* o freqüència de tall. Aquesta

freqüència es defineix com la freqüència de la referència sinusoidal tal que la corresponent sortida s'atenua - es redueix en amplitud - a un valor de  $1/\sqrt{2}$  vegades el valor de l'entrada, que equival a una atenuació de 3 dB. Per tant, el client especifica únicament aquesta freqüència de tall, i cal garantir que la resposta s'atenuï tal i com s'acaba d'indicar.

## 3.2 Estratègies de control i estructures emprades

En el moment d'iniciar la col·laboració, *Ingenia* dissenyava controladors PID en temps continu i que posteriorment discretitzava per tal de controlar els seus motors. A part de mantenir aquesta opció, es decideix provar diverses alternatives amb l'objectiu de comprovar si els resultats es poden millorar. Per tant, les opcions contemplades són les següents:

1. **Disseny en temps continu i posterior discretització:** es manté l'estructura PID, i a més es prova una variant d'aquesta, l'estructura I-PD, que presenta l'avantatge d'eliminar el sobrepic. Per a cadascuna d'aquestes estructures es calcularan els paràmetres del controlador, i s'obtindrà per discretització el controlador digital anàleg.
2. **Disseny en temps discret:** en general, la millor opció és fer el disseny del controlador en temps discret, i no pas com es contempla en el punt anterior. Així doncs, es provarà una estructura digital de tipus genèrica coneguda com a RST, que si bé prové d'un controlador PID - o variants -, la manipulació posterior causa que se'n perdi la interpretació com a tal.

Els subsegüents apartats detallen tot el procediment seguit per a cadascuna de les propostes.

### 3.2.1 Disseny en temps continu i discretització

#### Estructura PID

L'estructura clàssica PID es pot representar a través del següent diagrama de blocs:

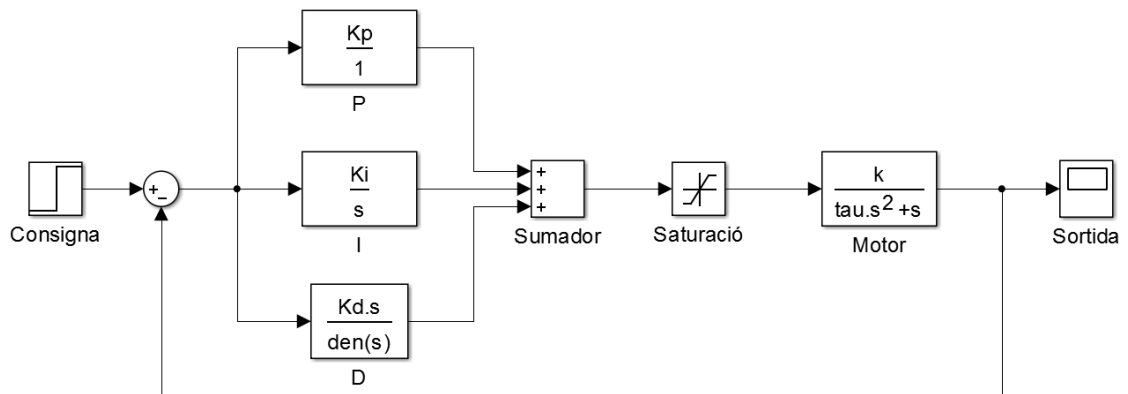


Figura 3.1: Estructura de control PID en temps continu

on  $den(s)$  al bloc D teòricament val 1, però *Simulink* requereix que tots els blocs siguin causals - l'ordre del polinomi denominador ha de ser més gran o igual que el del numerador. Per tant, per a mantenir el comportament del sistema alhora que es compleix el requeriment del *Simulink*, cal imposar tants pols despreciables - molt ràpids - com sigui necessari. En altres figures on aparegui  $den(s)$  es fa el mateix.

Per altra banda, es pot observar que s'incorpora un element saturador en l'esquema final. Es pretén representar així la limitació pròpia dels motors, en tant que si l'acció de control excedeix el valor màxim assumible pel motor, aquesta acció es limitarà al valor de saturació. Evidentment, aquest fet causarà que el motor deixi de comportar-se tal i com s'ha predit, afectant així la resposta del sistema. Com s'ha comentat anteriorment, un dels objectius del treball és l'estudiar la limitació dels motors en base al fenomen de la saturació; per tant, els controladors que es proposin no hauran d'excedir aquests valors de saturació.

La funció de transferència que correspon a la Figura 3.1 és:

$$W(s) = \frac{Y(s)}{R(s)} = \frac{k(K_d s^2 + K_p s + K_i)}{s^2(\tau s + 1) + k(K_d s^2 + K_p s + K_i)} \quad (3.1)$$

I el controlador PID un cop discretitzat presenta l'estructura següent:

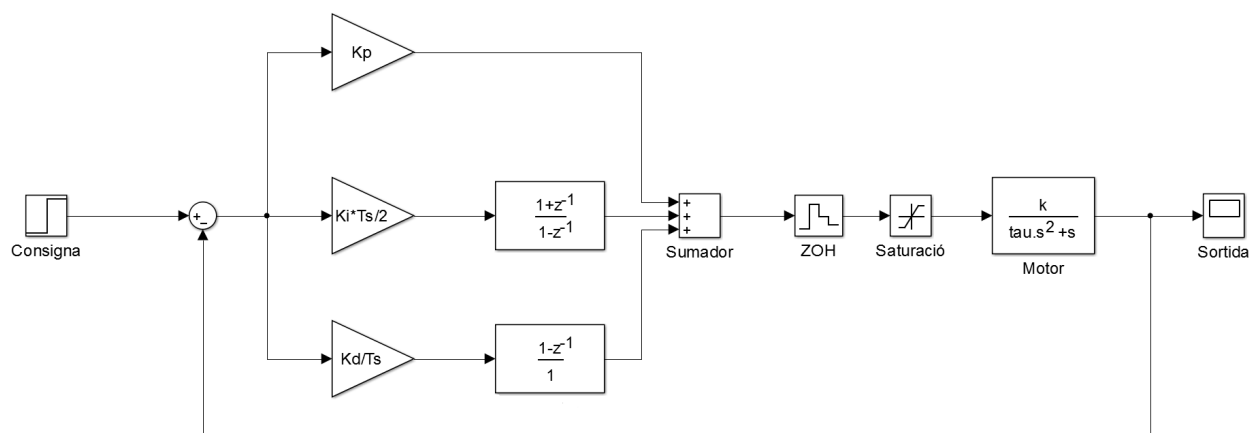


Figura 3.2: Estructura de control PID discretitzada

essent  $T_s$  el període de mostreig. No resulta necessari trobar l'expressió de la funció de transferència en llaç tancat per a la Figura 3.2 ja que tots els càlculs es realitzen en temps continu.

### Estructura I-PD

L'estructura I-PD constitueix una variant de l'estructura PID presentada anteriorment, millorant respecte de l'anterior els fets de no presentar sobrepuig al no introduir zeros addicionals en la funció de transferència en llaç tancat.

Aquesta estructura es pot representar a través del següent diagrama de blocs:

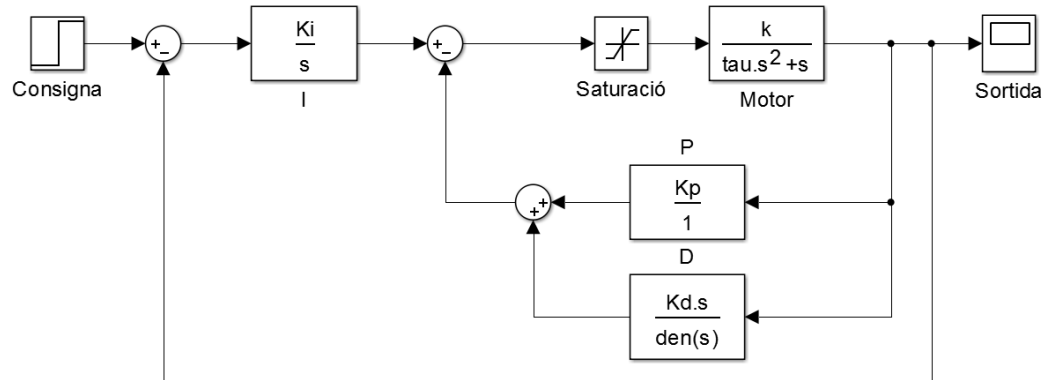


Figura 3.3: Estructura de control I-PD en temps continu

I la seva corresponent versió discretitzada és:

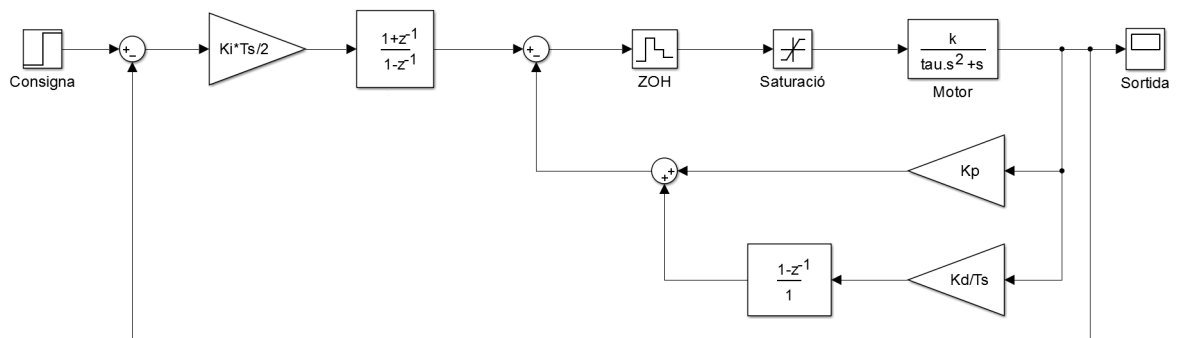


Figura 3.4: Estructura de control I-PD en temps discret

No obstant, es decideix treballar amb una variació de l'estructura presentada en la Figura 3.3: la nova versió - Figura 3.5 - és totalment equivalent pel que fa a funció de transferència en llaç tancat i permet, en cas que es desitgi, trobar-ne l'equivalent RST - explicada en l'apartat següent - de forma més senzilla.

Així doncs, l'estructura amb què es treballa és:

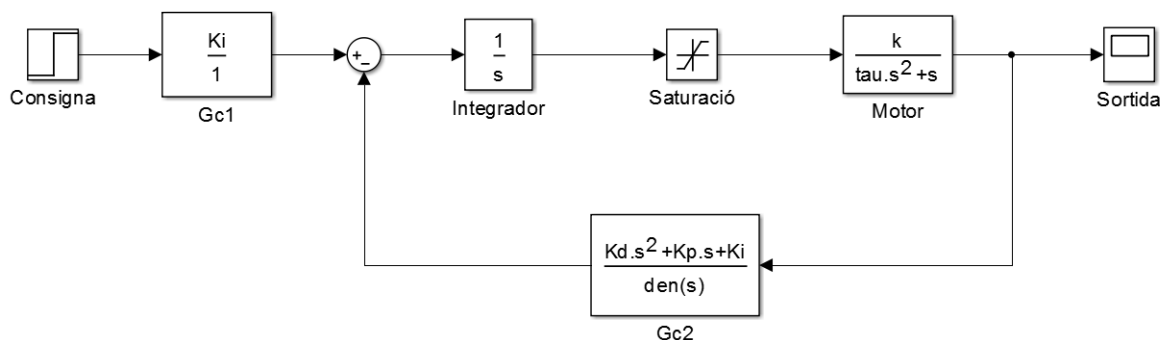


Figura 3.5: Variant de l'estructura de control I-PD en temps continu

La funció de transferència en llaç tancat per a les figures 3.3 i 3.5 amb què es treballarà és:

$$W(s) = \frac{Y(s)}{R(s)} = \frac{kK_i}{s^2(\tau s + 1) + k(K_d s^2 + K_p s + K_i)} \quad (3.2)$$

### 3.2.2 Disseny en temps discret

Fins ara s'ha tractat el cas en què es dissenyaven els controladors en temps continu i es discretitzaven per a obtenir-ne la corresponent versió en temps discret, que era directament implementada. Si bé és una possibilitat, la teoria suggereix que un procediment que proporciona millors resultats consisteix a dissenyar directament el controlador en temps discret - per exemple, pel fet de considerar des de l'inici l'efecte del mantenidor. Ara bé, també complica una mica el model en afegir un pol a la funció de transferència en llaç tancat, cosa que no succeeix si es procedeix de l'altra manera. Per tant, es prova també un control RST en temps discret amb l'objectiu de comprovar les diferències entre uns i altres resultats.

El mètode RST designa, tal i com es tracta en el capítol 3 de [8], l'estructura general *de tres branques* dels controladors digitals. Un controlador RST no és més que l'estructura polinomial que s'obté en manipular de forma algebraica les expressions que descriuen el controlador inicial, que pot ser de molts tipus diferents. Per a tots els tipus de controladors que es poden concebre com a punt de partida, l'estructura general RST és la mateixa, variant entre ells els valors dels paràmetres. Així doncs, es tria com a punt de partida un controlador PID digital obtingut per discretització del controlador PID continu, i es deriven totes les relacions necessàries per a



obtenir el model final del controlador RST.

Sigui

$$G_{PID}(s) = K \left[ 1 + \frac{1}{T_i s} + \frac{T_d s}{1 + \frac{T_d}{N} s} \right] \quad (3.3)$$

la funció de transferència d'un controlador PID continu, amb:

- $K$  - guany proporcional
- $T_i$  - acció integral
- $T_d$  - acció derivativa
- $T_d/N$  - filtrat de l'acció derivativa

Es pot discretitzar aquesta expressió fent servir, per exemple, l'aproximació en diferències cap endarrere, fent el canvi  $s = (1 - q^{-1})/T_s$ . Procedint així, l'expressió per al controlador PID discretitzat és:

$$G_{PID}(q^{-1}) = K \left[ 1 + \frac{T_s}{T_i} \cdot \frac{1}{1 - q^{-1}} + \frac{\frac{NT_s}{T_d + NT_s}(1 - q^{-1})}{1 - \frac{T_d}{T_d + NT_s}q^{-1}} \right] \quad (3.4)$$

Aquesta expressió es pot escriure de forma més compacta - en forma de fracció de polinomis  $R(q^{-1})$  i  $S(q^{-1})$  - si es sumen els tres termes de l'equació (Eq. 3.3). Siguin

$$R(q^{-1}) = r_0 + r_1 q^{-1} + r_2 q^{-2} \quad (3.5a)$$

$$S(q^{-1}) = (1 - q^{-1})(1 + s'_1 q^{-1}) = 1 + s_1 q^{-1} + s_2 q^{-2} \quad (3.5b)$$

el numerador i el denominador de  $G_{PID}(q^{-1})$ , respectivament, s'aconsegueix passar de la forma PID a la forma RS. El controlador és:

$$G_{PID}(q^{-1}) = \frac{R(q^{-1})}{S(q^{-1})} = \frac{r_0 + r_1 q^{-1} + r_2 q^{-2}}{1 + s_1 q^{-1} + s_2 q^{-2}} \quad (3.6)$$

I les relacions entre els paràmetres del controlador PID discret i del RST són:

$$r_0 = K \left( 1 + \frac{T_s}{T_i} - N \frac{T_s}{T_d} s'_1 \right) \quad (3.7a)$$

$$r_1 = K \left[ s_1 \left( 1 + \frac{T_s}{T_i} + 2N \frac{T_s}{T_d} s'_1 \right) - 1 \right] \quad (3.7b)$$

$$r_2 = -K s_1 \left( 1 + N \frac{T_s}{T_d} \right) \quad (3.7c)$$

$$s'_1 = -\frac{T_d}{T_d + NT_s} \quad (3.7d)$$

Nota: per a que es pugui obtenir l'equivalent controlador PID en temps continu a partir del de temps discret, s'ha de verificar  $-1 < s'_1 \leq 0$ .

Segui  $G_{planta}(q^{-1}) = \frac{B(q^{-1})}{A(q^{-1})}$ , es pot representar el conjunt planta-controlador a través del següent diagrama de blocs:

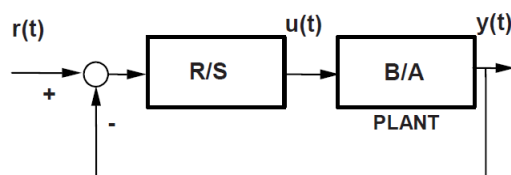


Figura 3.6: Conjunt planta-controlador en forma RS

Si es modifica convenientment, es pot representar el controlador amb l'estructura estàndard de tres branques:

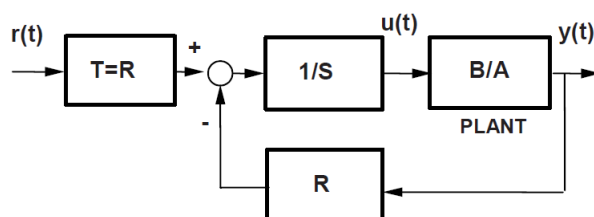


Figura 3.7: Conjunt planta-controlador en forma RST

Com es comenta en [8] (p. 100), imposant una sèrie de condicions, es troba que  $T(q^{-1}) =$

$R(1)$ ; és a dir,  $T(q^{-1})$  és un factor que permet preservar el guany unitari del sistema en llaç tancat.

La implementació final en *Simulink* és la següent:

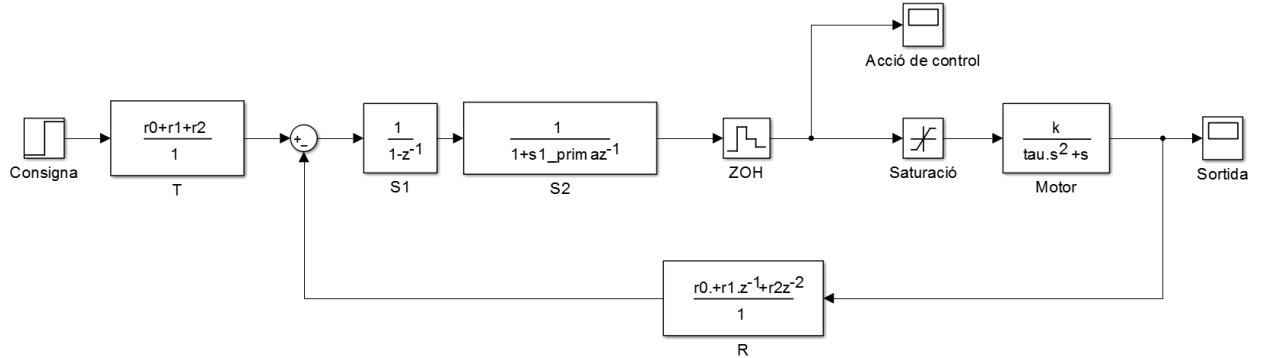


Figura 3.8: Implementació en *Simulink* del conjunt planta-controlador en forma RST

### 3.3 Determinació dels pols a partir del *bandwidth*

En el present treball es segueix el mètode d'assignació de pols, en què s'igualen els coeficients per graus del denominador de la funció de transferència del sistema en llaç tancat - que depèn dels paràmetres del controlador - i del denominador desitjat, que queda determinat pels pols que es vol que el sistema presenti. La posició d'aquests pols desitjats s'obté a partir de les especificacions demanades; en el present cas, el *bandwidth*. Per tant, es pot determinar la relació entre el *bandwidth* i els pols del sistema a partir de la funció de transferència en llaç tancat del sistema. És per això que en l'apartat anterior s'han presentat les estructures dels controladors que s'empraran.

En el capítol 8 de [9] es presenta el mètode general per a trobar el *bandwidth* d'un sistema -en temps continu -, que permet derivar relacions entre aquest *bandwidth* i els corresponents pols del sistema, però només es pot aplicar amb èxit per als casos més senzills. Per tant, com que cal determinar les constants  $K_p$ ,  $K_i$  i  $K_d$ , és necessari fixar tres pols. Es parteix de la opció més senzilla, que consisteix en forçar un únic pol amb multiplicitat triple; si aquesta tria no proporciona bons resultats es consideraran estratègies més complexes. Tot i així, en l'apartat de resultats es veurà com aquesta elecció és satisfactòria.

El procediment a seguir és el següent: s'aplica el mètode general per a trobar en temps

continu la relació entre pol i *bandwidth*, que servirà per als casos esmentats en l'apartat 3.2.1. A partir d'aquest resultat, però, es podrà trobar la relació equivalent per al cas del disseny en temps discret - apartat 3.2.2 -, fent ús també de la relació

$$pol_z = e^{pol_s T} \quad (3.8)$$

essent  $pol_s$  el pol en temps continu - trobat analíticament;  $pol_z$ , l'equivalent pol en temps discret, i  $T$  el període de mostreig.

### 3.3.1 Mètode general en temps continu

El mètode general per a relacionar el pol i el *bandwidth* consta dels següents passos:

1. Trobar la funció de transferència del sistema, i expressar-la en termes de la variable  $s$  a partir de la transformada de Laplace, si s'escau.
2. Fer el canvi  $s \rightarrow j\omega$ .
3. Trobar la magnitud de l'expressió obtinguda en el punt anterior.
4. Igualar l'expressió de la magnitud a  $\frac{1}{\sqrt{2}}$ , que s'obté per al cas  $\omega = \omega_{BW}$ , i resoldre l'equació resultant per trobar  $\omega_{BW}$ .

Com ja s'ha comentat anteriorment, s'escull que el sistema en llaç tancat presenti un únic pol amb multiplicitat tres. La forma més general és

$$G_{desitjat}(s) = \frac{1}{(\tau s + 1)^3} \quad (3.9)$$

Si es segueixen els passos anteriorment descrits, primer cal fer la substitució  $s \rightarrow j\omega$ :

$$G_{desitjat}(\omega) = \frac{1}{(j\tau\omega + 1)^3} \quad (3.10)$$

La magnitud de l'expressió anterior és:

$$|G_{desitjat}(\omega)| = \frac{1}{\left[\sqrt{\tau^2\omega^2 + 1}\right]^3} \quad (3.11)$$

Igualant l'equació (Eq. 3.11) a  $\frac{1}{\sqrt{2}}$  s'imposa  $\omega = \omega_{BW}$ :

$$\frac{1}{\left[\sqrt{\tau^2 \omega_{BW}^2 + 1}\right]^3} = \frac{1}{\sqrt{2}} \quad (3.12)$$

Fins ara no ha aparegut  $pol_s$  - pol desitjat en temps continu. Es relaciona amb  $\tau$  per mitjà de

$$pol_s = -1/\tau \quad (3.13)$$

Substituint l'equació (Eq. 3.13) en l'equació (Eq. 3.12) s'obté:

$$\frac{1}{\left[\sqrt{\frac{\omega_{BW}^2}{pol_s^2} + 1}\right]^3} = \frac{1}{\sqrt{2}} \quad (3.14)$$

Aïllant convenientment de l'equació (Eq. 3.14) s'obté que la relació final és:

$$pol_s = \pm \frac{\omega_{BW}}{\sqrt[3]{2} - 1} \quad (3.15)$$

Es selecciona la opció negativa per tal que el sistema sigui estable. Així, finalment:

$$pol_s = -\frac{\omega_{BW}}{\sqrt[3]{2} - 1} \quad (3.16)$$

### 3.3.2 Aplicació del resultat al cas RST

S'ha comentat anteriorment que el fet de fer el disseny en temps discret força l'aparició d'un pol extra. Per tant, per al mètode RST - disseny directament en temps discret - caldrà explicitar quatre pols. La solució més senzilla torna a ser la tria d'un únic pol, en aquest cas amb multiplicitat quàdruple. Per tant, es pot deduir a partir del cas anterior, i fent ús de l'equació (Eq. 3.8) que el pol desitjat en temps discret ve donat per

$$pol_z = e^{-\frac{\omega_{BW}}{\sqrt[4]{2} - 1}T} \quad (3.17)$$

### 3.4 Assignació de pols a través del codi desenvolupat

El següent pas, un cop es té clara la relació entre l'especificació en *bandwidth* i el pol corresponent, és aplicar pròpiament el mètode d'assignació de pols. Com que les expressions que es manipulen són llargues, resulta tediós desenvolupar a mà tot el procediment. Per a simplificar la computació, s'elaboren codis en MATLAB que resolen de forma simbòlica el procés d'assignació de pols, presentant l'avantatge addicional de poder recrear diversos escenaris d'interès; per exemple, per a trobar el límit de prestacions dels motors abans d'entrar en saturació.

En els subsegüents apartats es presenten breument els mètodes i passos que permeten obtenir els paràmetres per a tots els controladors estudiats. Per a trobar la descripció completa de tot el procés cal referir-se als codis C.2.1 i C.2.2. de l'annex.

#### 3.4.1 Controladors PID i I-PD

El codi desenvolupat per a calcular les constants dels controladors PID i I-PD és el mateix per a tots dos casos, i es pot trobar a l'annex C.2.1. Això es deu al fet que els denominadors de les funcions de transferència enllaç tancat - equacions (Eq. 3.1) i (Eq. 3.2) - coincideixen per a ambdues estructures. Siguin

$$den_{actual}(s) = s^2(\tau s + 1) + k(K_d s^2 + K_p s + K_i) \quad (3.18a)$$

$$den_{desitjat}(s) = (\tau s + 1)^3 \quad (3.18b)$$

el denominador enllaç tancat i el denominador desitjat, respectivament, l'assignació de pols per igualació de coeficients proporciona les següents relacions:

$$K_p = \frac{3\tau \cdot pol_s^2}{k} \quad (3.19a)$$

$$K_i = \frac{-\tau \cdot pol_s^3}{k} \quad (3.19b)$$

$$K_d = \frac{-3\tau \cdot pol_s - 1}{k} \quad (3.19c)$$

Introduint la relació obtinguda en l'equació (Eq. 3.16) es pot comprovar com afecta el *bandwidth* especificat als valors dels guany dels controladors.

### 3.4.2 Controlador RST

El desenvolupament de fórmules generals per al controlador RST presenta més complexitat que el cas anterior en tant que les expressions algebraiques amb què cal treballar presenten més dificultat de manipulació.

En l'apartat 3.2.2 de [8] es detalla tot el procediment que cal seguir fins a obtenir el sistema d'equacions final. El procediment és idèntic al del cas anterior: s'igualen els coeficients del denominador desitjat - costat esquerre de la igualtat - amb els coeficients actuals - costat dret. Novament, el codi de l'annex C.2.2 il·lustra tot el procés.

Introduint els canvis

$$a'_1 = a_1 - 1 \quad (3.20a)$$

$$a'_2 = a_2 - a_1 \quad (3.20b)$$

$$a'_3 = -a_2 \quad (3.20c)$$

per a facilitar - lleugerament - la manipulació de les expressions, i essent  $b_1$  i  $b_2$  els coeficients del numerador de la planta, el sistema d'equacions final és:

$$-4pol_z = b_1 r_0 + s'_1 + a'_1 \quad (3.21a)$$

$$6pol_z^2 = b_2 r_0 + b_1 r_1 + s'_1 a'_1 + a'_2 \quad (3.21b)$$

$$-4pol_z^3 = b_2 r_1 + b_1 r_2 + s'_1 a'_2 + a'_3 \quad (3.21c)$$

$$pol_z^4 = b_2 r_2 + s'_1 a'_3 \quad (3.21d)$$

De nou, si es fa ús de l'equació (Eq. 3.17), es poden relacionar els paràmetres del controlador RST amb el *bandwidth* del sistema.

Tant per al disseny de temps continu com de temps discret s'obté la solució paramètrica del sistema d'equacions, depenent aquesta únicament d'una variable, el *bandwidth*. La gràcia d'aquesta forma de procedir és que es poden recrear fàcilment tots els casos que es desitgin estudiar únicament variant el valor d'aquesta variable.

### 3.5 Límit de prestacions dels motors

Com s'ha comentat amb anterioritat, existeixen diverses consideracions que cal tenir en compte en dissenyar controladors. Així, aquests factors poden causar que l'acció de control no presenti la resposta predita. Un dels factors més importants i que constitueix l'objecte d'estudi del capítol és la saturació.

La saturació, tractada en [14], és un fenomen que limita el valor màxim d'una magnitud física - en aquest cas la tensió - que el motor pot assumir en el seu règim normal de funcionament. Per tant, si es supera el llindar marcat per aquest valor de saturació, el valor aplicat al motor disminueix fins al valor de saturació. Això provoca que la resposta del motor difereixi de la resposta que es pot obtenir en simulació si el model no inclou els efectes d'aquest fenomen.

Per tal d'abordar aquest problema, es vol garantir que l'acció de control dissenyada no superi en cap instant el valor de saturació - malgrat que seria tolerable una lleugera saturació. Per tant, el que cal estudiar és que el valor màxim de l'acció de control - a partir d'ara, *valor de pic* - no superi aquest valor llindar. Aquest problema s'estudia sobre l'estructura I-PD pels següents motius:

- No es realitza l'estudi en temps discret ja que la complexitat de manipulació en aquest cas és elevada.
- Pel que s'acaba de comentar, s'ha de fer en temps continu. Per tant, es tria l'estructura que a priori es faria servir, que és la I-PD pels avantatges que presenta enfront de l'estructura PID.

El diagrama de blocs convenientment reordenat - respecte de la Figura 3.5 - per a mostrar amb més claredat l'acció de control com a sortida del sistema és:



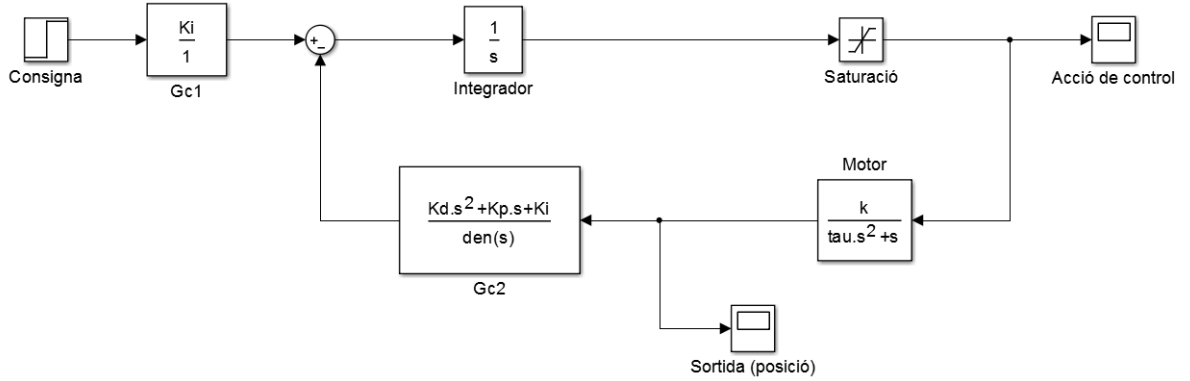


Figura 3.9: Diagrama de la variant I-PD per a l'estudi de la saturació

I la funció de transferència de la Figura 3.9 en què es pren com a sortida l'acció de control és:

$$W_{sat}(s) = \frac{Y(s)}{R(s)} = K_i \frac{s(\tau s + 1)}{s^2(\tau s + 1) + k(K_d s^2 + K_p s + K_i)} \quad (3.22)$$

Aïllant la sortida de l'equació (Eq. 3.22) se n'obté la seva expressió:

$$Y(s) = R(s)W_{sat}(s) = K_i \frac{A(\tau s + 1)}{s^2(\tau s + 1) + k(K_d s^2 + K_p s + K_i)} \quad (3.23)$$

essent  $R(s) = \frac{A}{s}$  la consigna graó en el domini  $s$ .

El que cal fer ara és clar: cal trobar el valor de pic de  $Y(s)$  en funció del *bandwidth*, de forma que es pugui estudiar quin és el valor màxim de *bandwidth* que es pot donar com a especificació abans que actui la saturació. Per a fer-ho, cal prendre l'equació (Eq. 3.23), aplicar l'antitransformada de Laplace - previ pas a fraccions simples - per a obtenir  $Y(t)$  i trobar-ne el màxim a partir d'igualar  $Y'(t)$  a zero. Com es pot observar, és un procés feixuc - i per al RST és pitjor encara -, per la qual cosa es fa ús del *software* Maple, que permet realitzar fàcilment càlculs amb variables simbòliques sense declarar-les, motiu pel qual se'l prefereix en aquest apartat per sobre del MATLAB. Un cop es té l'expressió del valor de pic s'importa aquest resultat al MATLAB.

Allà es dissenya un algorisme que calcula el valor de pic tot fent un escombrat de *bandwidth* fins que es troba - de forma gràfica - el valor llindar. Per a aquest valor llindar, es representa

gràficament la resposta i es verifica que el valor de pic coincideix amb el valor de saturació.

En la memòria únicament es presenten, a títol il·lustratiu, les figures finals, fruit de tot aquest estudi tan laboriós, sobre el motor del Laboratori. Per a més detalls, es poden trobar aquest estudi a l'annex C.2.3.

Així doncs, la primera figura permet determinar - ampliant-la degudament - el valor límit de *bandwidth*. Aquest  $\omega_{limit}$  és el valor per al qual coincideixen el valor de saturació - que és de 5 volts per al motor del Laboratori - i el valor de pic.

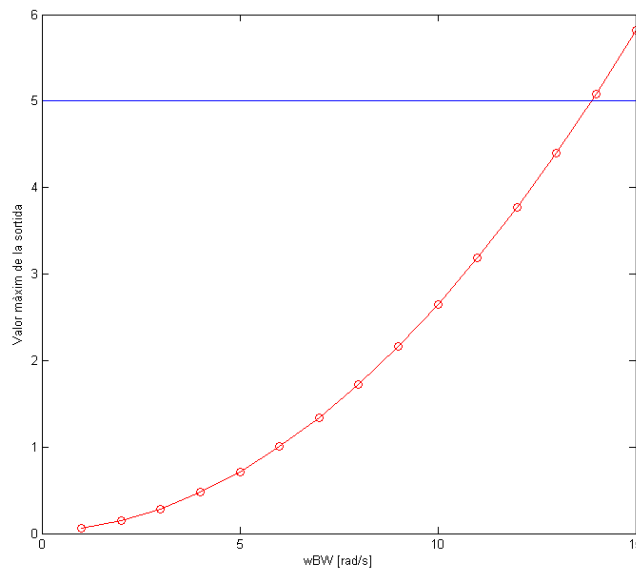


Figura 3.10: Escombrat en *bandwidth* i valors de pic corresponents

S'obté un valor màxim  $\omega_{limit} = 13.88$  rad/s. Si es substitueix aquest valor en l'equació (Eq. 3.23) i es grafica, la sortida presenta el següent aspecte:

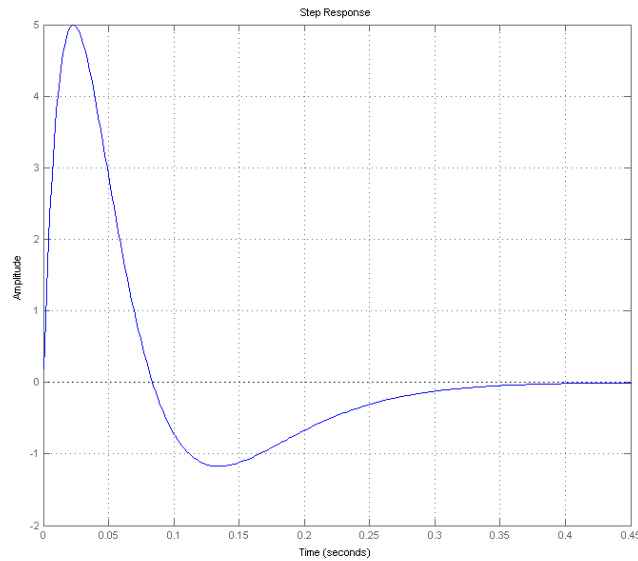


Figura 3.11: Acció de control per al cas de  $\omega_{limit}$

Es pot comprovar com, efectivament, el valor de pic de l'acció de control és el valor de saturació per al cas  $\omega = \omega_{limit}$ . Per tant, per al cas del motor del Laboratori, sembla que no es podran demanar valor de *bandwidth* superiors a 13.88 rad/s.

### 3.6 Resultats del control

Finalment, en aquest capítol es presenten els resultats de tots els controladors provats. El conjunt de proves que s'han dut a terme es resumeix en la següent llista:

1. Motor del Laboratori
  - (a) Simulació: estructures PID, I-PD i RST.
  - (b) Experiments: estructures PID, I-PD i RST.
2. Motor d'*Ingenia*
  - (a) Simulació: estructures PID, I-PD i RST.
  - (b) Experiments: únicament l'estructura PID, ja que és l'única implementable en aquest motor en tant que el *firmware* només està desenvolupat per a aquesta estructura.

A continuació es detallen tots i cadascun d'aquests casos, i s'analitzen degudament en l'últim punt del capítol. Es presenten en primer lloc el valor del *bandwidth* estudiat i les corresponents constants obtingudes per als controladors: es recorda, tal i com es diu en l'apartat 3.4.1, que aquests valors són iguals per als casos PID i I-PD. Posteriorment es poden observar les diverses gràfiques obtingudes tant en simulació com de forma experimental, en què la funció blava correspon a una consigna gràó; i la vermella, a la corresponent sortida del motor.

### 3.6.1 Motor del Laboratori

El motor del Laboratori de l'ETSEIB es fa servir a les pràctiques de l'assignatura de Control amb Computador, i és per això que disposa d'una interfície que facilita la comunicació amb el motor. Per a més informació sobre com treballar amb aquest motor es pot consultar [4], el document oficial per a les pràctiques de l'assignatura anteriorment esmentada.

## Simulació

El model emprat per a obtenir els resultats en simulació és el que es va presentar en l'apartat 2.3.1 i que provenia d'un experiment previ per a determinar un model de referència del motor.

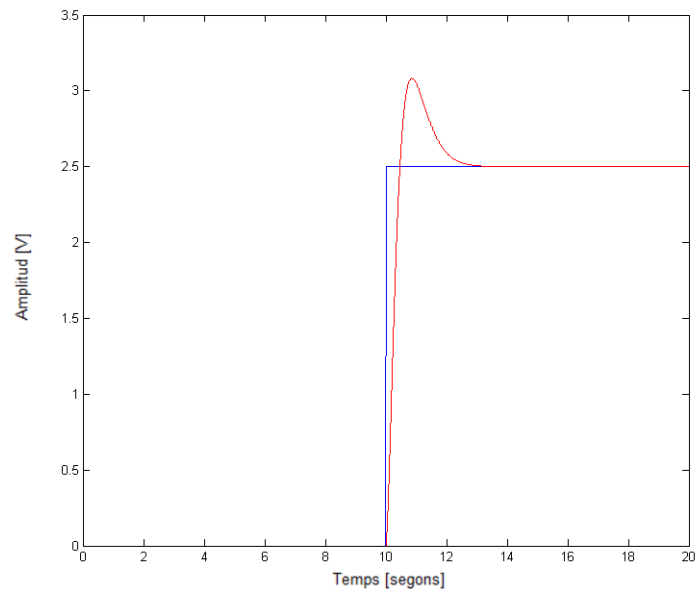


Figura 3.12: Resposta del motor amb un controlador PID

$$w_{BW} = 1.75 \text{ rad/s}$$

$$K_p = 0.9819$$

$$K_i = 1.1234$$

$$K_d = 0.1935$$

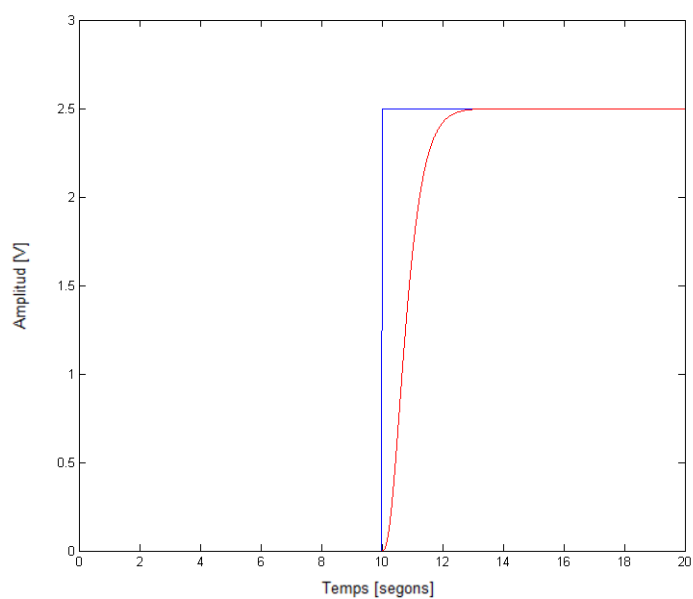


Figura 3.13: Resposta del motor amb un controlador I-PD

$$w_{BW} = 1.75 \text{ rad/s}$$

$$K_p = 0.9819$$

$$K_i = 1.1234$$

$$K_d = 0.1935$$

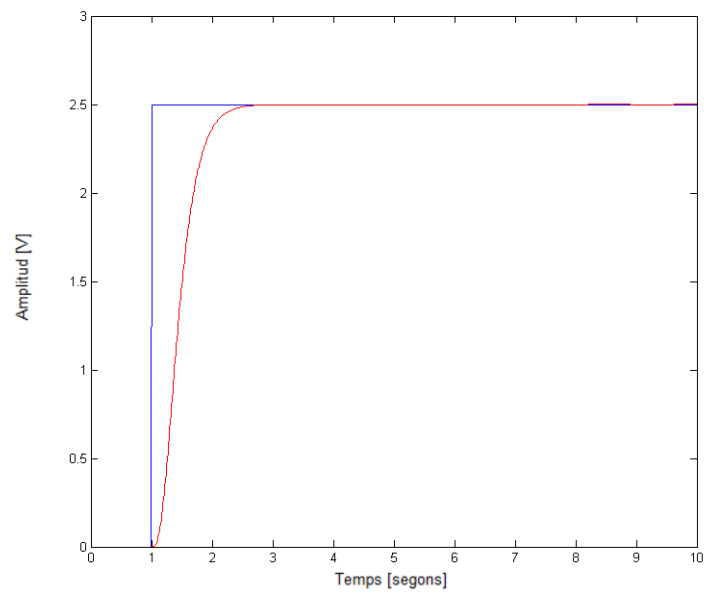


Figura 3.14: Resposta del motor amb un controlador RST

$$w_{BW} = 3.5 \text{ rad/s}$$

$$r_0 = 11.129$$

$$r_1 = -21.465$$

$$r_2 = 10.351$$

$$s'_1 = -0.668$$

## Experiments

Com a comentari previ, en realitzar els experiments el motor parteix d'una posició que no es correspon amb un nivell de tensió 0, i per això fa un primer procés transitori.

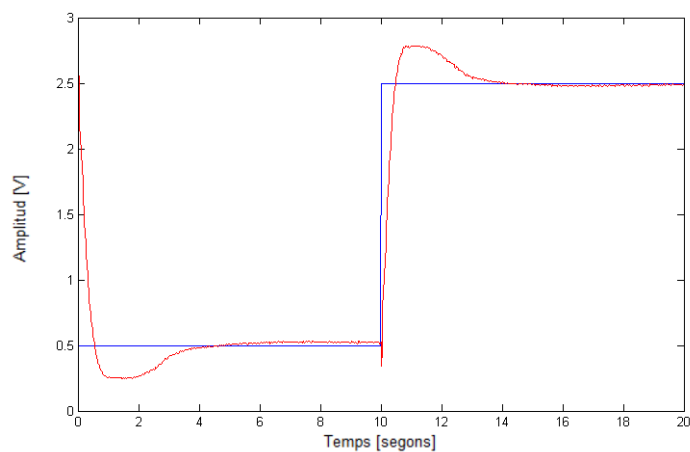


Figura 3.15: Resposta del motor amb un controlador PID

$$w_{BW} = 1.75 \text{ rad/s}$$

$$K_p = 0.9819$$

$$K_i = 1.1234$$

$$K_d = 0.1935$$



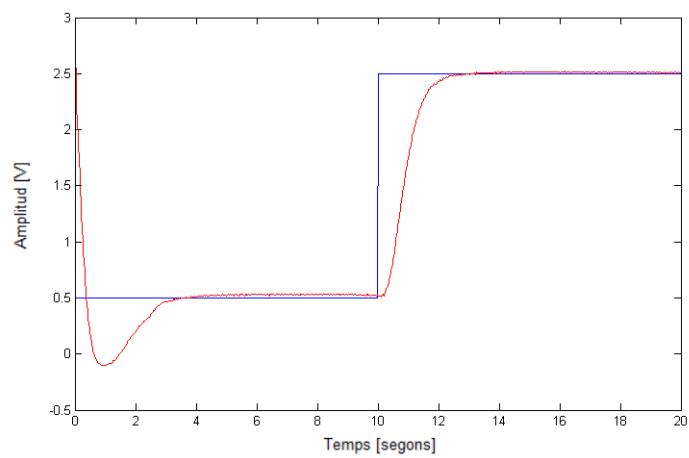


Figura 3.16: Resposta del motor amb un controlador I-PD

$$w_{BW} = 1.75 \text{ rad/s}$$

$$K_p = 0.9819$$

$$K_i = 1.1234$$

$$K_d = 0.1935$$

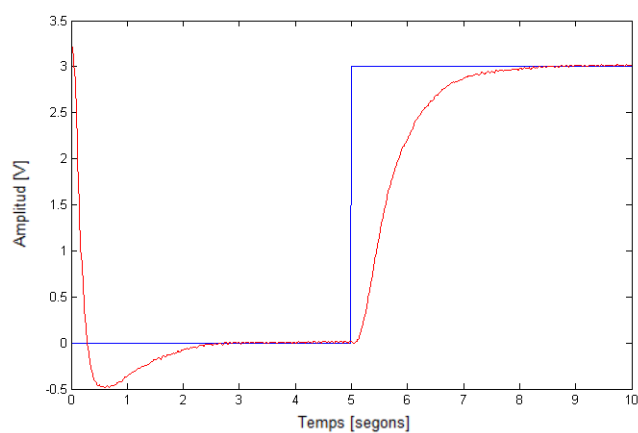


Figura 3.17: Resposta del motor amb un controlador RST

$$w_{BW} = 3.5 \text{ rad/s}$$

$$r_0 = 11.129$$

$$r_1 = -21.465$$

$$r_2 = 10.351$$

$$s'_1 = -0.668$$

### 3.6.2 Motor d'*Ingenia*

Per al cas del motor d'*Ingenia*, aquest funciona una mica diferent que el motor del Laboratori pel que fa a la interpretació de les magnituds d'entrada i de sortida del motor. Es pot observar que s'aplica una consigna - *position demand* o posició demanada - de 1000. Aquella posició demanada està relacionada amb el parell demanat al motor, que s'expressa en ‰ del parell nominal: aquest fet es deu a que hi ha un llaç de parell per sota del llaç de posició que s'encarrega de garantir que el motor proporciona el parell que se li demana. Per tant, especificant 1000 com a posició demanada, s'està demanant el parell nominal del motor.

Pel que fa a la sortida - *position actual value* o posició mesurada -, aquesta s'expressa en funció de les comptes del conversor analògic digital o *CAD*. Aquests motors tenen un llaç de realimentació de la posició, que genera una tensió proporcional a la posició en la que es troba; aquesta tensió entra al microcontrolador *Pluto* i es quantifica mitjançant l'anteriorment esmentat *CAD*.

Un cop indicat com s'han d'interpretar els valors dels senyals, es pot procedir a presentar-ne els resultats obtinguts, tant en simulació com experimentalment.

## Simulació

Per a obtenir els resultats en simulació s'empra el model obtingut en llaç obert amb la *toolbox ident*, presentat en l'apartat 2.3.3.

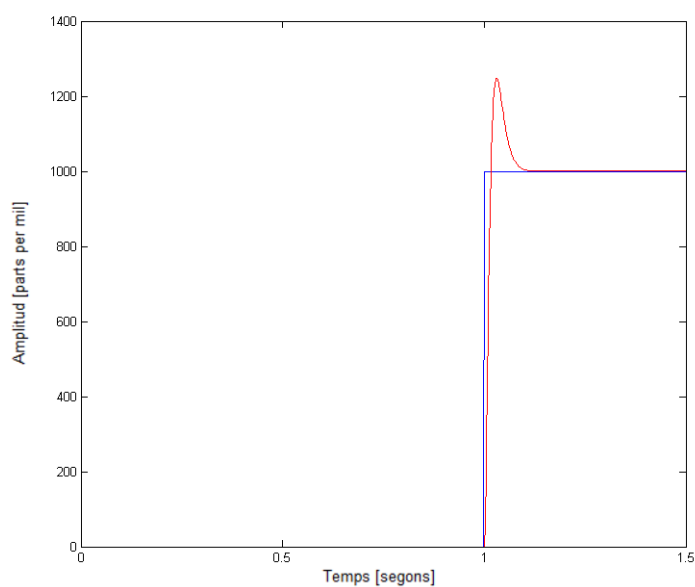


Figura 3.18: Resposta del motor amb un controlador PID

$$w_{BW} = 50 \text{ rad/s}$$

$$K_p = 0.1497$$

$$K_i = 4.8931$$

$$K_d = 0.0009$$

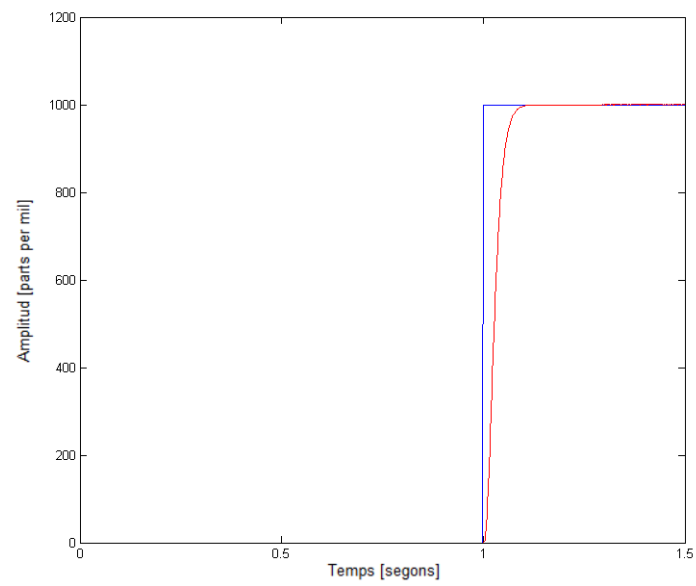


Figura 3.19: Resposta del motor amb un controlador I-PD

$$w_{BW} = 50 \text{ rad/s}$$

$$K_p = 0.1497$$

$$K_i = 4.8931$$

$$K_d = 0.0009$$

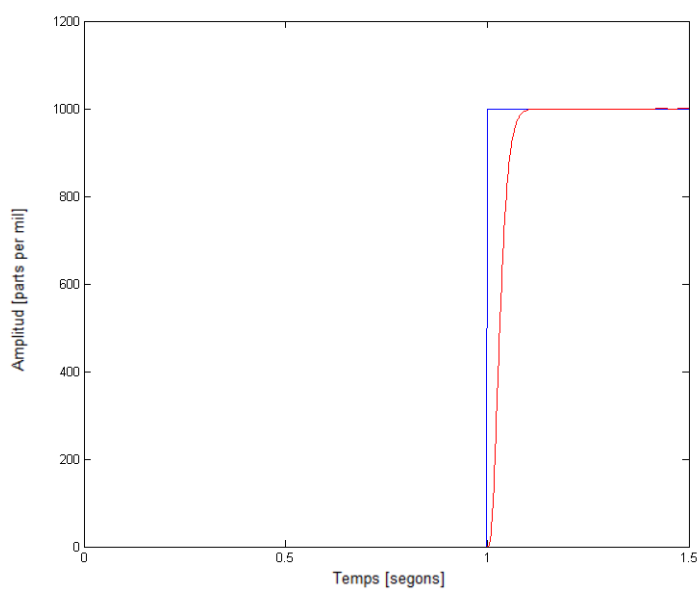


Figura 3.20: Resposta del motor amb un controlador RST

$$w_{BW} = 50 \text{ rad/s}$$

$$r_0 = 0.1893$$

$$r_1 = -0.3508$$

$$r_2 = 0.1623$$

$$s'_1 = -0.6931$$

## Experiments

En primer lloc, cal comentar que s'han d'aplicar uns factors d'escalatge a les constants calculades a través dels codis de MATLAB. El seu objectiu és el de reduir el cost computacional i augmentar la resolució, per la qual cosa cal introduir el període de mostreig  $T_s = 0.001$  s i un factor d'escalat  $K_{esc} = 2^{16}$ , que prové de l'ús d'un quantificador amb resolució de 16 bits per a representar la informació. Per tant, introduint aquests dos paràmetres, les correccions que cal aplicar són:

$$K'_p = K_p \cdot K_{esc}$$

$$K'_i = K_i \cdot T_s \cdot K_{esc}$$

$$K'_d = \frac{K_d \cdot K_{esc}}{T_s}$$

on els guanys que apareixen aïllats a l'esquerra de la igualtat són els que cal aplicar realment al motor - els altres són els calculats pel MATLAB. Per a ampliar aquesta informació es pot consultar [6].

Un cop es té en compte aquest fet, es manté l'especificació en *bandwidth* i es repeteixen els experiments fets en simulació. Ara bé, tal i com s'ha comentat prèviament, només es pot provar l'estructura PID, que és la que està implementada actualment en els programes d'*Ingenia*. El comportament del motor es fa inestable, generant-se vibracions i sorolls. Per tant, aquest controlador no té l'efecte desitjat.

Per tal d'aïllar el problema, es repeteix l'experiment amb un simple controlador proporcional. Els resultats obtinguts en simulació són:

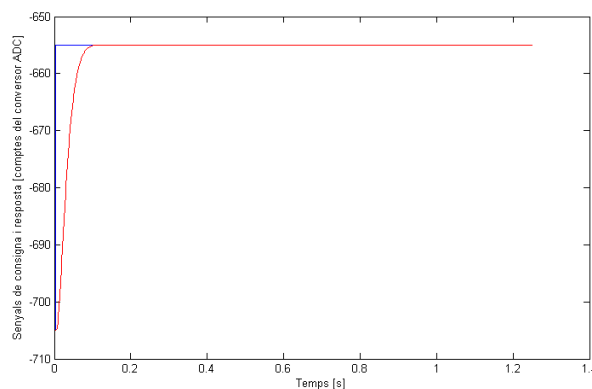


Figura 3.21: Resultats en simulació amb un controlador P

I els experimentals són:

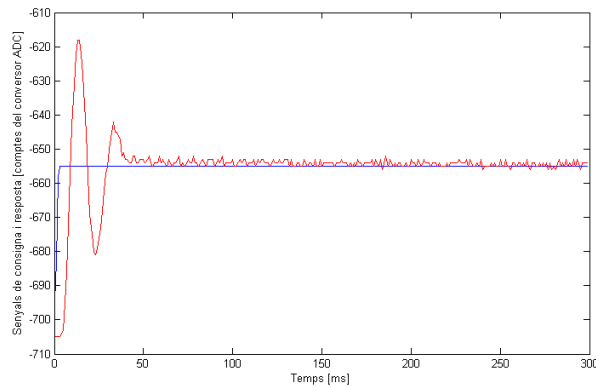


Figura 3.22: Resultats experimentals amb un controlador P

S'observa una clara diferència entre ambdues figures deguda a que el model del motor no explica prou bé el seu comportament real. Per tant, es desenvolupa un model alternatiu, que té en compte un altre fenomen no lineal que no s'havia contemplat fins al moment: la zona morta, la qual es posa de manifest en observar el motor necessita una certa tensió per arrencar, ja que ha de vèncer una fricció inicial important.

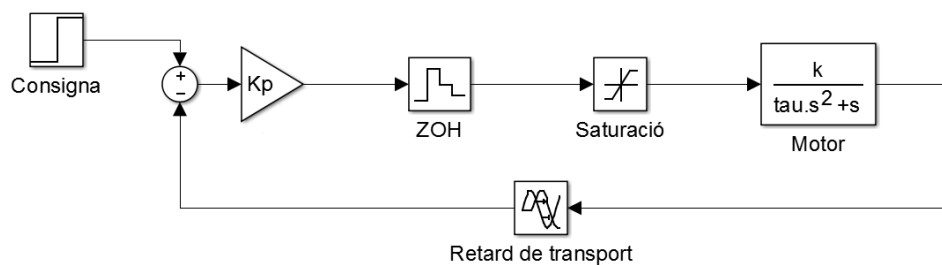


Figura 3.23: Model modificat per al motor d'*Ingenia*

Es modela inicialment aquest fenomen de zona morta com a retard de transport, i s'ajusta manualment el nombre de mostres de retard fins que els resultats en simulació s'aproximen prou als experimentals, mostrats a la Figura 3.22. Amb  $N = 25$  mostres de retard, els resultats en simulació són els següents:



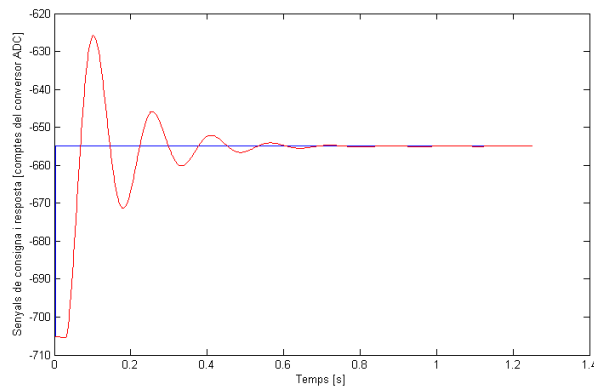


Figura 3.24: Resultats en simulació amb un controlador P i model amb zona morta

S'observa com el nou model - Figura 3.23 - explica molt millor el comportament real del motor. No obstant, encara s'haurà d'estudiar amb més detall per a millorar aquesta primera proposta.

### 3.6.3 Resum dels resultats

Es presenten a continuació els resultats obtinguts per als controladors, tal i com es va fer en l'apartat 2.4.4 per als diversos models identificats.

	Estructures PID i I-PD		Estructura RST	
	Especificació	Paràmetres	Especificació	Paràmetres
<b>Motor del Laboratori</b>	$w_{BW} = 1.75 \text{ rad/s}$	$K_p = 0.9819$ $K_i = 1.1234$ $K_d = 0.1935$	$w_{BW} = 3.5 \text{ rad/s}$	$r_0 = 11.129$ $r_1 = -21.465$ $r_2 = 10.351$ $s'_1 = -0.668$
<b>Motor d'Ingenia</b>	$w_{BW} = 50 \text{ rad/s}$	$K_p = 0.1497$ $K_i = 4.8931$ $K_d = 0.0009$	$w_{BW} = 50 \text{ rad/s}$	$r_0 = 0.1893$ $r_1 = -0.3508$ $r_2 = 0.1623$ $s'_1 = -0.6931$

Taula 3.1: Resum dels diversos controladors obtinguts

### 3.7 Conclusions sobre el control

En primer lloc, es sintetitza tot el procediment seguit: s'han dissenyat controladors en temps continu - estructures PID i I-PD modificada - i en temps discret - RST - amb l'objectiu de veure si hi ha diferències significatives en els resultats proporcionats. Cal tenir en compte, però, que l'estat actual dels programes d'*Ingenia* només permet implementar la versió PID del controlador. A més, es desitja incloure en el model l'efecte de la saturació en el comportament dels motors.

Analitzant els resultats obtinguts, es pot veure com la hipòtesi prèvia sobre les millores en la resposta segons el tipus de controlador emprat sembla correcta. El controlador PID continu original presenta sobrepuig, fet que es corregeix amb el controlador continu I-PD - lleugerament modificat. I encara més, el controlador RST discret manté aquesta cancel·lació del sobrepuig i permet assolir especificacions en *bandwidth* més exigents tot mantenint una bona resposta.

Ara bé, els casos que s'han presentat constitueixen els casos límit de bon funcionament per al motor del Laboratori. Per a especificacions en *bandwidth* per sobre d'aquests valors però encara per sota del valor límit teòric calculat, el motor vibra i es comporta de forma inestable. Caldria estudiar amb més detall per què no s'obtenen bons resultats en aquests casos, tot i que això es deu molt possiblement al fet que el convertidor de potència de la planta no permet arribar-hi, i no pas per fenòmens no considerats en el model del motor.

Pel que fa al motor d'*Ingenia*, els experiments amb un controlador PID no han permès obtenir resultats satisfactoris, i aquests suggereixen que el model concebut per als motors no és capaç de copsar-ne prou bé el seu caràcter. En el present treball només s'ha considerat la saturació com a fenomen no lineal, però s'intueix que poden haver-hi altres fenòmens no lineals no capturats en el model; en concret, un comportament amb zona morta segons la qual el sistema no respon a l'entrada fins que no es superi la fricció estàtica. Per a verificar-ho es genera un nou model que contempli aquesta zona morta, modelitzada en primera instància com a retard de transport, i s'executa de nou la simulació. Com es pot observar en comparar les figures 3.21 a 3.24, aquest nou model representa millor el comportament del motor. És lògic, doncs, que els controladors dissenyats sobre el primer model no funcionin com s'espera, ja que aquell model no cospa bé tots els fenòmens que regeixen el comportament del motor.

Per tant, cal estudiar més a fons aquest fenomen de zona morta amb l'objectiu de trobar un model per al motor encara millor sobre el qual dissenyar totes les diverses estructures de control, les quals hauran de permetre obtenir-ne una millor resposta.

## Conclusions

Aquest capítol de conclusions no pretén repetir l'anàlisi dels resultats d'identificació i control, que ja s'ha realitzat al final de cadascun dels respectius capítols. En canvi, el seu objectiu és el de donar indicacions generals en base a l'experiència adquirida al llarg de la realització d'aquest projecte.

Pel que fa a la identificació a partir de l'entrada i la sortida del sistema, és interessant tractar de deduir prèviament l'estructura - els ordres - del model en base a les seves equacions. Com que això no sempre és possible, es pot seguir un mètode de prova i error, provant ordres i estructures diverses fins a trobar la que s'adapti millor a les dades.

Un altre aspecte molt important és el fet d'eliminar l'efecte de l'integrador de les dades sempre que es conegui prèviament la seva presència: s'ha comprovat que aquest fet millora substancialment els resultats obtinguts en l'ajust dels models. Es recorda que va ser necessari realitzar aquest procediment per al motor d'*Ingenia* ja que, a diferència del motor del Laboratori, no disposava de sensor de velocitat, havent de mesurar la posició i comportant aquest fet l'aparició d'un integrador pur.

Pel que fa al control, és interessant destacar la millora que s'obté en la resposta del motor si es passa de l'estructura PID original a les alternatives I-PD i RST. Ara bé, al contrari del que es pensava a l'inici, les solucions obtingudes amb controladors PID i I-PD dissenyats en temps continu i posteriorment discretitzats proporcionen resultats prou bons si es comparen amb els obtinguts amb el controlador RST dissenyat directament en temps discret. No obstant això, el controlador RST permet l'acompliment d'especificacions en *bandwidth* més exigents que no pas les seves alternatives.

Però possiblement la conclusió més important que es pot extreure d'aquest treball és la necessitat de desenvolupar un nou model que inclogui fenòmens no lineals que aquest treball només ha apuntat al final de l'estudi, i que podrien explicar el perquè de les discrepàncies entre els resultats obtinguts en simulació i els observats de forma experimental quan simplement s'han

provat especificacions en *bandwidth* inferiors al valor límit teòric, per a les quals els resultats en simulació difereixen dels resultats experimentals. En el present treball només s'ha estudiat la saturació com a no linealitat, però sembla no ser suficient: experiments finals van suggerir la presència d'una zona morta en el comportament del motor. Incloent aquest fenomen i provant un control molt senzill, els resultats obtinguts eren molt millors que els obtinguts anteriorment.

Per tant, la continuació natural d'aquest projecte és un estudi més detallat del comportament no lineal que presenta el motor d'*Ingenia* per tal de millorar el model dinàmic. Aquest motor respon tal i com es preveu quan ja es troba en moviment, però hi ha un fenomen no modelat en l'arrencada que causa aquesta disparitat de resultats: si s'aconsegueix modelar bé aquest fenomen, els controls que es desenvolupin seran més efectius que no pas els actuals. Es pot anticipar que s'hauran d'usar tècniques de control predictiu, com ara el predictor de Smith (secció 8.2 de [1]), de forma que es pugui modelar aquest fenomen de retard per separat i dissenyar els controladors emprant les mateixes tècniques que fins al moment.

## Bibliografia

- [1] ÅSTRÖM, K., AND HÄGGLUND, T. *Control PID Avanzado*, 2009.
- [2] CAO, Y., WU, S., AND JIAO, Z. Design and simulation of voice coil motor for the micro-electric load simulator. *School of Automation Science and Electrical Engineering, Beihang University, China* (2011).
- [3] DALIFELD. <http://dali.feld.cvut.cz/ucebna/matlab/toolbox/fdident/msinclip.html>, 2005.
- [4] ESAII. *Treballs pràctics de Control Digital*.
- [5] FENG, X., DUAN, Z., FU, Y., AND SUN, A. The technology and application of voice coil actuator. *The Academy of Military Transportation, Tianjin, China* (2011).
- [6] INGENIAMC. <http://doc.ingeniamc.com/display/EMCL/General+manufacturer+objects#expand-0x2500Positioncontrolparametersset>, 2013.
- [7] INGENIAMC. <http://doc.ingeniamc.com/display/PLU/Product+Features>, 2013.
- [8] LANDAU, I., AND ZITO, G. *Digital Control Systems: Design, Identification and Implementation*, 2006.
- [9] LEE, T. H. *The Design of CMOS Radio-Frequency Integrated Circuits*, 2004.
- [10] LJUNG, L. *System Identification Toolbox For Use with MATLAB*, 2002.
- [11] MATHWORKS. <http://es.mathworks.com/help/ident/ref/oe.html>, 2014.
- [12] NATIONALINSTRUMENTS. <http://www.ni.com/white-paper/4028/en/>, 2010.
- [13] VERHAEGEN, M., AND VERDULT, V. *Filtering and System Identification: A Least Squares Approach*, 2007.

- [14] VUKOSAVIC, S. N. *Digital Control of Electrical Drives*, 2007.

## A Pressupost

El pressupost considerat per a reproduir el present projecte és el següent:

Concepte	Cost [€]
Planta del Laboratori	4500
Motor de <i>Bei Kimco Magnetics</i>	240
<i>Pluto DC Servo Drive</i>	600
Font d'alimentació DC (24v)	145
Llicència del <i>software</i> MATLAB®	1500
Llicència del <i>software</i> Maple®	1500
PRESSUPOST D'EXECUCIÓ	8485
Despeses generals (5%)	424,25
Honoraris	10250
<b>SUBTOTAL</b>	<b>19159,25</b>
IVA (21%)	4023,44
<b>TOTAL</b>	<b>23182,69</b>

Val a dir que la taula anterior recull els preus que hauria de pagar un usuari que desitgés reproduir el projecte. En el present cas d'estudi, l'equip ja disposava de totes les llicències de *software* necessàries, i pel que fa als motors i altres elements físics, tant *Ingenia* com l'Escola van cedir aquests elements a cost zero.

Pel que fa als honoraris, s'ha considerat un sou i un nombre d'hores de dedicació aproximats. S'estima que el director del projecte, a qui se li assigna un sou de 50€/hora, ha dedicat unes 70 hores al projecte; pel que fa a l'estudiant, es contempla un sou de 15€/hora i una dedicació total de 450 hores al projecte.





## B Estudi d'impacte ambiental

L'aplicació del mètode presentat en aquest treball per a la identificació i control de motors de tipus *voice coil* tindrà més impacte a nivell econòmic que no pas ambiental, tot i que es poden dur a terme algunes accions per tal de minimitzar els efectes que el projecte pugui causar al medi ambient.

En desenvolupar codis de sintonia per als controladors, els quals proporcionen directament les constants del controlador en base a l'especificació en *bandwidth* del sistema, es redueix el temps de còmput del problema de control. Es preveu que aquest estalvi en temps tingui un impacte econòmic positiu.

L'impacte mediambiental d'aquest projecte serà petit en tant que el consum d'energia - principalment la necessària per alimentar el motor - és molt baix, i la vida útil dels motors és suficientment llarga com per a no esdevenir un problema. Tot i així, aquest impacte es pot reduir si els motors es reciclen adientment quan acaba la seva vida útil: es poden desmuntar i classificar els seus components en funció dels materials, permetent així la recuperació i reaprofitament dels materials valuosos que encara puguin fer servei. No es genera cap tipus de residu tòxic, fum, etc., que calgui tractar. És per això que l'impacte mediambiental del projecte serà mínim.



## C Codis

En aquest apartat s'inclouen els codis que s'han citat al llarg del treball, però no són els únics que s'han desenvolupat. En el CD annex es poden trobar tots els codis que s'han creat per a poder fer els diversos estudis, convenientment classificats.

Nota: la compilació del document dóna errors si es mantenen els accents en els comentaris dels codis. Per tant, tot i incórrer en faltes d'ortografia en eliminar-los, es mantenen els comentaris per a facilitar la comprensió dels codis.

### C.1 Codis per a la identificació

#### C.1.1 Generació del senyal multisinus

```
%CODI: ingeniaMCmsin

A = 1; % defineix amplitud maxima
np = 1; % numero de periodes
ts = 0.0001; %definicio ts

f = 20:20:4000;
f = logspace(log10(20),log10(4000), 20);
f = round(f.*0.5)/0.5;
fr = f

%f=10:1:4000; % de la 10 a la 1000 de 10 en 10
w = 2.*pi.*f; % pas a pulsacions
[cx] = msinclip2(f,ones(1,length(f))); % optimitzacio per CF minim
temps = ts:ts:np/f(1); % generacio vector temps
entrada = zeros(1,length(temps)); % entrada a 0
```

```

for i=1:length(cx) % bucle que omple entrada amb valors de cosinus
    entrada = entrada+abs(cx(i)).*cos(w(i).*temps+angle(cx(i)));
end

entrada = entrada-mean(entrada); % centra a 0 el valor mig
entrada = entrada.*A/max(abs(entrada)); % amplitud maxima a A
entrada = round(entrada*32768);
plot(temps,entrada)
xlabel('Time')
ylabel('Multisine amplitude')
dlmwrite('Multisine.txt',[entrada])

[gy, fy] = tdf(entrada,fr,ts,0);
semilogx(w,20*log10(gy))
grid on

```

### C.1.2 Codi d'ajust amb estructura OE

```

% CODI: Ajust_traient_integrador

Ts = 0.001;
Lectura_variables;

% Planta trobada amb toolbox ident

% channels_data (1r experiment)
K = 1390.3;
tau = 7.1867e-03;

% %channels_data2 (2n experiment)
% K = 1295;
% tau = 6.4063e-03;

Gmotor = tf([K], [tau 1])

% Definicio de freqüències i DFT dels senyals

fr = 2:2:400;
wr = fr*2*pi;
Fs = 1/Ts;

```

```

[gu, fu] = tdf(u, fr, Ts, 0);
[gy, fy] = tdf(y, fr, Ts, 0);

% Model i representacio

Gmotorz = c2d(Gmotor, Ts)
[mag, phase] = bode(Gmotorz,wr);
[mags, phases] = bode(Gmotor,wr);

figure(1)
title('Resposta Frecuencial Motor');
bodemag(Gmotorz,'r')

grid on
hold on
bodemag(Gmotor,'b')
semilogx(wr, 20*log10(gy./gu),'r*','MarkerSize',3)

% Test identificacio temporal del model
% -----

Z = [y u];

% Model Output Error

OE111_T = oe(Z, [1 1 1], 'Ts', Ts)
GOE111t = tf(OE111_T.B, OE111_T.F, Ts)

figure(2)
bodemag(GOE111t)
title('Bode Identificacio Temporal');
hold on

% Dibuixar magnitud mesures
semilogx(wr, 20*log10(gy./gu),'r*','MarkerSize',3)

gum = gu.*cosd(fu)+sqrt(-1).*gu.*sind(fu);
gym = gy.*cosd(fy)+sqrt(-1).*gy.*sind(fy);
H = gym./gum;

hold on

```

```

bodemag(Gmotorz, 'g') % Bode del model teoric

% Test identificacio frequencial del model
% -----

ZFD = iddata(gym,gum,'ts',Ts,'Domain','Frequency','FREQ',wr);

OE111_F = oe(ZFD, [1 1 1], 'Ts', Ts)
GOE111f = tf(OE111_F.B, OE111_F.F, Ts)

figure(3)
bodemag(GOE111f, 'b');

hold on
bodemag(Gmotorz, 'g')
title('Bode Identificacio Frequencial');
hold on

% Dibuixar magnitud mesures
semilogx(wr, 20*log10(gy./gu), 'r*', 'MarkerSize', 3)

```

## C.2 Codis per al control

### C.2.1 Solució dels controladors PID i I-PD continus

```

% Codi: Continu_PolTriple_1_EqsGenerals

f=1000;
T=1/f;

% Triar la planta: Motor Lab Control / Motor Ingenia

% Parametres del motor del laboratori de Control

k = 1.02*1.62/(9*0.017);
tau = 0.3;

% % Parametres del motor d'Ingenia
% kesc=1;
% k = 68.01/kesc;

```

```

% tau = 7.9119e-3;

Gp = tf([k],[tau 1 0]);

% Determinacio de les constants del controlador per assignacio
% de pols. EXPLICACIO DEL PROCEDIMENT:

% El polinomi caracteristic desitjat es:

% pol_caract = poly[pol pol pol], amb:

% pol = -1*wBW/sqrt((2)^(1/3)-1) (aplicant la definicio de BW a
% la planta).

% S'obte: pol_caract = 1 +p1*s^(-1) +p2*s^(-2) +p3*s^(-3), on:

% p1 = p(1,2); % p1 = -3*pol
% p2 = p(1,3); % p2 = 3*(pol^2)
% p3 = p(1,4); % p3 = -(pol^3)

% Resolucio simbolica del sistema en funcio de wBW i polaux,
% escrivint les equacions utilitzades en tots els passos.

syms pol wBW Kp Ki Kd
[Kp, Ki, Kd, pol] = solve('-3*pol = (1+k*Kd)/tau', '3*(pol^2) = k*Kp/tau',
'-(pol)^3 = k*Ki/tau', 'pol = -1*wBW/sqrt((2)^(1/3)-1)', 'Kp', 'Ki', 'Kd'
, 'pol')

% Un cas d'estudi

wBW_triat = 5;

Ki_triada = double(subs(eval(Ki), wBW, wBW_triat))
Kp_triada = double(subs(eval(Kp), wBW, wBW_triat))
Kd_triada = double(subs(eval(Kd), wBW, wBW_triat))

```

### C.2.2 Solució del controlador RST discret

```

% CODI: Discret_1

```

```
f = 1000;
T = 1/f;

% % Parametres del motor del Laboratori de Control
% k = 1.02*1.62/(0.017*9);
% tau = 0.30;

% Parametres del motor d'Ingenia
k = 1656;
tau = 8.59e-3;

Gs = tf([k], [tau 1 0])
Gz = c2d(Gs, T)

[Nz,Dz] = tfdata(Gz,'v'); % planta en forma de vectors

b1 = Nz(2);
b2 = Nz(3);
a1 = Dz(2);
a2 = Dz(3);

% Canvi de notacio per seguir la del llibre:

a1_prima = a1-1;
a2_prima = a2-a1;
a3_prima = -a2;

% Determinacio de les constants del controlador per assignacio
% de pols. EXPLICACIO DEL PROCEDIMENT:

% El polinomi caracteristic desitjat es:
% pol_caract = poly[polz polz polz polz], amb:

% polz = exp(-1*(wBW/sqrt(2^(1/4)-1))*T) (aplicant la definicio de BW a
% la planta).

% S'obtindra: pol_caract = 1 +p1*z^(-1) +p2*z^(-2) +p3*z^(-3)
% +p4*z^(-4), on:

% p1 = p(1,2); % p1 = -4*polz
```



```

% p2 = p(1,3); % p2 = 6*polz^2
% p3 = p(1,4); % p3 = -4*polz^3
% p4 = p(1,5); % p4 = polz^4

% Resolucio simbolica del sistema en funcio de wBW, escrivint les equacions
% utilitzades en tots els passos.

syms r0 r1 r2 s1_prima polz K Ti Td Kp Ki Kd wBW
[r0, r1, r2, s1_prima, K, Ti, Td, TdN, Kp, Ki, Kd, polz] = solve
('-4*polz = b1*r0+s1_prima+a1_prima',
'6*polz^2 = b2*r0+b1*r1+s1_prima*a1_prima+a2_prima',
'-4*(polz)^3 = b2*r1+b1*r2+s1_prima*a2_prima+a3_prima',
'(polz^4) = b2*r2+s1_prima*a3_prima',
'K = (r0*s1_prima -r1 -r2*(2+s1_prima))/(1+s1_prima)^2',
'Ti = T*K*(1+s1_prima)/(r0+r1+r2)',
'Td = T*(s1_prima*r0 -s1_prima*r1 +r2)/(K*(1+s1_prima)^3)',
'TdN = -T*s1_prima/(1+s1_prima)', 'Kp=K', 'Ki=K/Ti', 'Kd=K*Td',
'polz = exp(-1*(wBW/sqrt(2^(1/4)-1))*T)', 'r0', 'r1', 'r2', 's1_prima',
'K', 'Ti', 'Td', 'TdN', 'Kp', 'Ki', 'Kd', 'polz')

% Estudi d'un cas particular:

wBW_triat = 3.5;

r0_triat = double(subs(eval(r0), {wBW}, {wBW_triat}))
r1_triat = double(subs(eval(r1), {wBW}, {wBW_triat}))
r2_triat = double(subs(eval(r2), {wBW}, {wBW_triat}))
s1_prima_triat = double(subs(eval(s1_prima), {wBW}, {wBW_triat}))

```

### C.2.3 Codi per a trobar el *bandwidth* límit

```

% CODI: Continu_PolTriple_3_LimitsSaturacio

% Aquí volem obtenir el valor maxm per a les K's del controlador que fan
% que no s'assoleixi la saturacio.
% Objectiu: trobar per a quin valor de wBW el valor de pic supera el valor
% de saturacio.
% Farem un bucle donant valors a wBW. Primer calcularem Ki i
% l'instant de pic, i amb tot això ja podrem calcular el maxm,
% substituint-ho tot plegat a l'expressio de la sortida.

```

```

% Quan la sortida sobrepassi la saturacio, haurem trobat un limit per a
% wBW. Amb aquest valor de wBW_limit, calculem les K's limit.

Continu_PolTriple_1_EqsGenerals;

A = 1; % amplitud grao
Ts = 0.001;

% Definir limits d'estudi:

wBW_inf = 1;
wBW_sup = 15;
pas = 1;

% Crear vectors on anirem emmagatzemant el valors de wBW, sortida i instant
% de pic.

wBW_estudiats = zeros((wBW_sup-wBW_inf)/pas+1, 1);
pics = zeros((wBW_sup-wBW_inf)/pas+1, 1); % guardar pic per a cada wBW
temps_pics = zeros((wBW_sup-wBW_inf)/pas+1, 1); % guardar temps de pic
% per a cada wBW

i = 1;

for wBW_triat = wBW_inf:pas:wBW_sup

    wBW_estudiats(i) = wBW_triat;

    pol_triat = double(subs(pol, wBW, wBW_triat));

    Ki_triada = double(subs(eval(Ki), wBW, wBW_triat));

    t_pic = -(2*tau*pol_triat+sqrt(2*((tau)^2)*((pol_triat)^2)
        +2*tau*pol_triat+1)+1)/(pol_triat*(tau*pol_triat+1)); % Maple
    temps_pics(i) = t_pic;

    pic = A * Ki_triada * exp(-((2 * tau * pol_triat) +
        sqrt((2 * tau ^ 2 * pol_triat ^ 2 + 2 * tau * pol_triat + 1)) + 1)
        / (tau * pol_triat + 1)) * (-2 * ((2 * tau * pol_triat) +
        sqrt((2 * tau ^ 2 * pol_triat ^ 2 + 2 * tau * pol_triat + 1)) + 1)
        / pol_triat / (tau * pol_triat + 1) + ((2 * tau * pol_triat) +

```

```

        sqrt((2 * tau ^ 2 * pol_triat ^ 2 + 2 * tau * pol_triat + 1)) + 1)
        ^ 2 / (pol_triat ^ 2) / (tau * pol_triat + 1) / tau) / 2;
        % ja se li substitueix directament t_pic
    pics(i) = pic;

    i = i+1;

end

% Escrivim en una matriu: [wBW|pic|temps_pic]' i grafiquem

dades = [wBW_estudiats pics temps_pics]';

figure;
plot(dades(1,:), dades(2,:), '-or')
saturacio = 5;
refline(0, saturacio); % wBW_limit (motor Laboratori) = 13,8778 rad/s
%title('Relacio entre el wBW i el valor de pic de la sortida');
xlabel('wBW [rad/s]');
ylabel('Valor maxim de la sortida')

wBW_limit = 13.8778;
pol_limit = double(subs(pol, wBW, wBW_limit)) % pol = f(wBW)

Ki_limit = double(subs(eval(Ki), wBW, wBW_limit))
Kp_limit = double(subs(eval(Kp), wBW, wBW_limit))
Kd_limit = double(subs(eval(Kd), wBW, wBW_limit))

% Plot de la sortida

num = (Ki_limit/tau)*[tau 1 0];
den = poly ([pol_limit pol_limit pol_limit]);
H = tf(num, den)

figure;
step(H)
grid on

```



## D Guia d'usuari del motor d'*Ingenia*

L'objectiu d'aquest apartat és el de proporcionar al lector una breu guia per tal que pugui reproduir els experiments amb el motor d'*Ingenia* que s'han dut a terme en aquest treball. La seva finalitat és, doncs, similar a la de [4].

### D.1 Generació de dades per a la identificació

A més de proporcionar el conjunt motor-controlador, *Ingenia* també ha elaborat un conjunt d'elements de *software* i *firmware* per a facilitar la realització dels experiments, i que es llisten tot seguit.

**\*.hexenc** els elements amb aquesta extensió corresponen al *firmware* - bloc d'instruccions - que es desitja carregar en cada cas. Es disposa de *firmware* preparat per a aplicar una estructura de control PID.

**FirmwareLoader** és un *software* que permet carregar els arxius de *firmware* disponibles, segons sigui el cas que es vol estudiar.

**\*.xdc** els elements amb aquesta extensió contenen la informació que determina quina és l'estructura del model que s'està carregant, principalment. En general, es crea un arxiu amb el llaç obert, i si es desitja es pot tancar tot especificant més endavant les constants del controlador.

**MotionLab** és el *software* que permet realitzar pròpiament l'experiment. Permet configurar

Així doncs, els experiments es duen a terme seguint aquests passos:

1. A través del *FirmwareLoader* es carrega l'element *\*.hexenc* que es vol estudiar. A continuació s'envia aquesta informació al *Pluto*.

2. Fent ús del *MotionLab*, es carrega l'element *\*.xdc* adequat i s'envien els canvis al *Pluto* fent un *Download to Drive*. A continuació es pot especificar el tipus de funció que es vol aplicar com a excitació al motor, així com modificar els seus paràmetres: val a dir que en el moment de fer els experiments només es pot variar l'amplitud d'una funció sinusoidal definida per punts. A més, cal definir la posició central de l'oscil·lació i configurar la durada de l'experiment. Finalment, si es desitgen realitzar experiments en llaç tancat, es poden definir les constants del controlador que tanca el llaç.

Cada vegada que es du a terme un experiment, els resultats es guarden en un full de càlcul. D'entre tots els paràmetres enregistrats, aquests són els especialment rellevants:

**Time** s'expressa en milisegons. El període de mostreig que es fa servir és de 1 *ms*; i la durada de l'experiment, de 500 *ms*.

**Position actual value** indica la posició actual del sistema expressada en termes del conversor ADC.

**Torque demand** mesura la senyal d'entrada al llaç de corrent. Equival a la sortida del llaç de control de posició més la *disturbance*.

El motor s'identifica a partir dels senyals que rep a l'entrada i que allibera a la seva sortida. En un diagrama de blocs, el bloc que representa el motor rep com a senyal d'entrada el *torque demand*; com a senyal de sortida, el *position actual value*.

## D.2 Aplicació dels controladors dissenyats

Per a aplicar els controladors dissenyats en MATLAB sobre el motor, cal fer ús novament del *software* desenvolupat per *Ingenia*. El procediment a seguir és el mateix que en l'apartat anterior: carregar el *firmware* que es correspongui amb l'estructura de control dissenyada amb el *FirmwareLoader*, i després definir les característiques de l'experiment emprant el *MotionLab*.

La diferència amb l'apartat anterior és que ara cal aplicar aquests controladors dissenyats. Això es pot fer a la pestanya *Motion* i configurant els guanys proporcional, integral i derivatiu a l'apartat *Loop Configuration*. Es recorda que cal aplicar un escalat addicional als valors dels guanys calculats, i que es poden consultar en [6]. Un cop aquests valors són introduïts, ja es pot executar l'experiment, obtenint un fitxer amb els resultats adientment enregistrats per a la seva anàlisi.

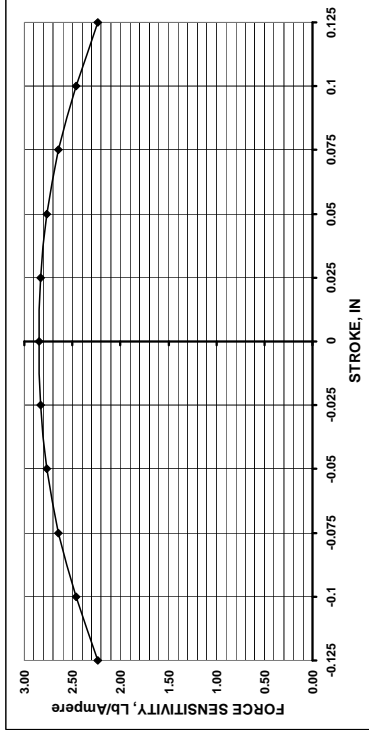
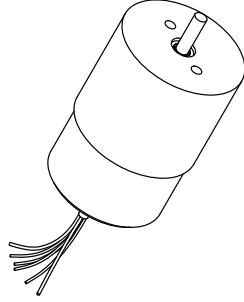
Winding Constants *	Units	Tol	Symbol	Wdg A
DC Resistance	Ohms	$\pm 12.5\%$	R	4.7
Voltage @ $F_p$	Volts	Nominal	$V_p$	33.0
Current @ $F_p$	Amps	Nominal	$I_p$	7.02
Force Sensitivity	LB/Amp	$\pm 10\%$	$K_f$	2.85
	N/Amp	$\pm 10\%$		12.68
Back EMF Constant	V/(Hz/sec)	$\pm 10\%$	$K_b$	3.86
	V/(mil/sec)	$\pm 10\%$		12.68
Inductance ***	milli-Henry	$\pm 30\%$	L	1.25

Linear Actuator Parameters *	Units	Symbol	Value
Peak Force **	LB	F <sub>P</sub>	20.0
	N		89.0
Continuous Stall Force **	LB	F <sub>CS</sub>	5.5
	N		24.47
Actuator Constant	LB/√in <sup>3</sup>	K <sub>A</sub>	1.31
	N/√in <sup>3</sup>		5.83
Electrical Time Constant	milli-sec	T <sub>E</sub>	270
Mechanical Time Constant	milli-sec	T <sub>M</sub>	1.51
Theoretical Acceleration	ft/sec <sup>2</sup>	a <sub>T</sub>	5647.9
	m/sec <sup>2</sup>		1721.5
Max Theoretical Frequency @ Full Stroke and Shasol® Triangular Motion	Hz	f <sub>max</sub>	117.2/130.2
Power FR @ F <sub>P</sub>	Watts	P <sub>P</sub>	232
Stroke:	± in		0.125
	± mm		3.18
Clearance on Each side of Coil	in		0.018
	mm		0.46
Thermal Resistance of Coil in still air	°C/Watt	θ <sub>JA</sub>	5.0
Maximum Allowable Coil Winding Temp	°C	Temp	155
Weight of Coil Assembly	OZ	WT <sub>C</sub>	1.82
Total Weight	g	WT <sub>T</sub>	51.7
	g		10.5
			297

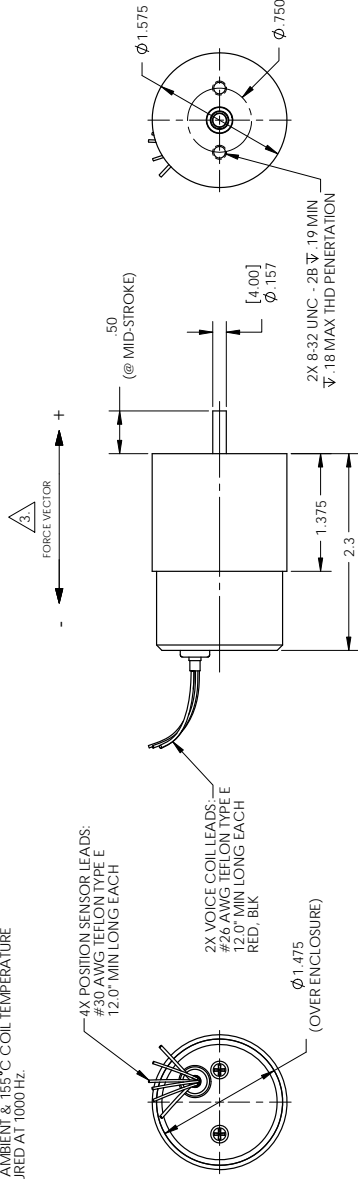
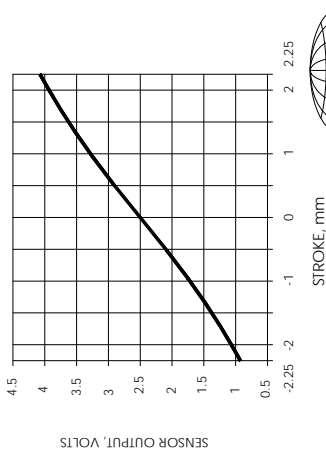
\* AT MID-STROKE POSITION AND @ 25°C AMBIENT TEMPERATURE  
 \*\* 10 SECONDS @ 25°C AMBIENT & 155°C COIL TEMPERATURE  
 \*\*\* @25°C AMBIENT & 155°C COIL TEMPERATURE  
 \*\*\*\* MEASURED AT 1000 HZ.

\* AT MID-STROKE POSITION AND @ 25°C AMBIENT TEMPERATURE  
\*\* 10 SECONDS @ 25°C AMBIENT & 155°C COIL TEMPERATURE

\*\*\* @25°C AMBIENT & 155°C COIL TEMPERATURE  
\*\*\*\* MEASURED AT 1000 HZ.



POSITION SENSOR		
LEAD WIRE	IDENTIFICATION	DESCRIPTION
YELLOW	$V_{cc}$	INPUT VOLTAGE (5 VOLTS)
GRAY	GND	GROUND
BROWN	$V_o$	OUTPUT VOLTAGE
WHITE	$V_{pp}$	VOLTAGE FOR PROGRAMMING ONLY NOT TO BE USED BY CUSTOMER

POSITION SENSOR ( $\pm 2.7\%$  MAXIMUM LINEARITY ERROR)

Proprietary rights of BEI Kinco are involved in the subject matter of this material and all manufacturing, reproduction, use, and sales pertaining to such subject matter are expressly reserved. This confidential and proprietary document is submitted for a specified purpose and the recipient by accepting this material agrees that this material will not be used, copied or reproduced in whole or in part nor its contents revealed in any manner or by any person except to meet the purpose for which it was delivered.

### THIRD ANGLE PROJECTION

UNLESS OTHERWISE SPECIFIED:  
 -ALL DIMENSIONS ARE IN INCHES  
 -BREAK SHARP EDGES .015 MAX  
 -SURFACE ROUGHNESS .63 ✓  
 -DIMENSIONS APPLY AFTER FINISH  
 -MAX FILLET R.010  
 -DIAMETERS SHALL NOT EXCEED  
 RUNOUT OF .005 FIM

**TOLERANCES:**

DECIMALS	ANG
.X $\pm .03$	$\pm 0$
.XX $\pm .01$	
.XXX $\pm .005$	

DO NOT SCALE DRAWING

**DO NOT SCALE DRAWING**

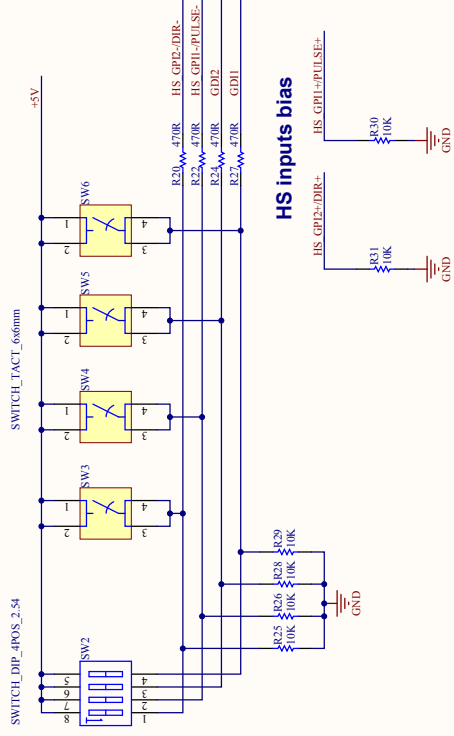
**BEL** KIMCO MAGNETICS DIVISION  
VISTA, CA 92081

VISTA, CA 92081

# LINEAR ACTUATOR SYSTEM

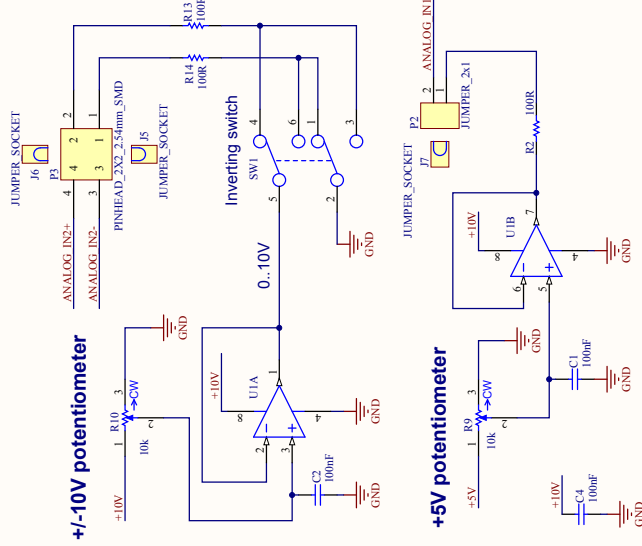
FILE NO.	L/PTOP/LAS	10/05/10	SIZE	FSCM NO.	DWG NO.	REV
		APPD GODKIN	C	55789	LAS16-23-000A	X1
		SCALE: NONE		SHEET: 1 OF 1		

# DIP switch and pushbuttons for digital inputs



HS inputs bias

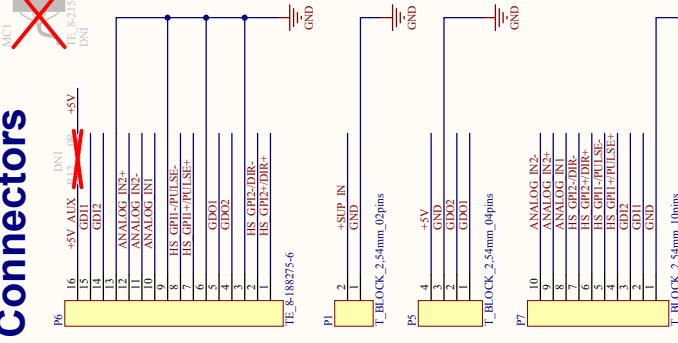
# Analog input potentiometers



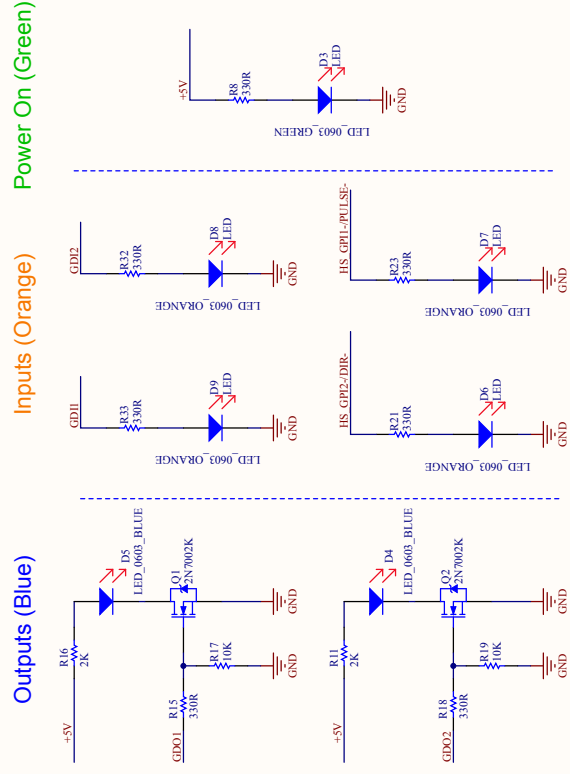
+/-10V potentiometer

+5V potentiometer

# Connectors



# Signalling LEDs

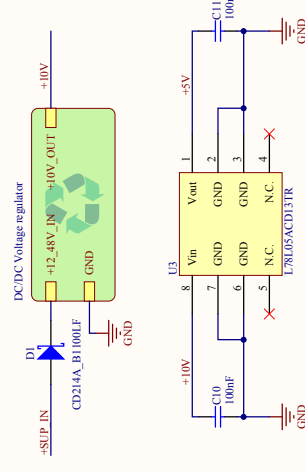


Outputs (Blue)

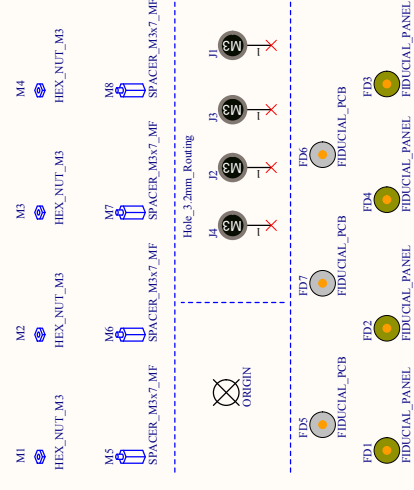
Inputs (Orange)

Power On (Green)

# Internal power supply



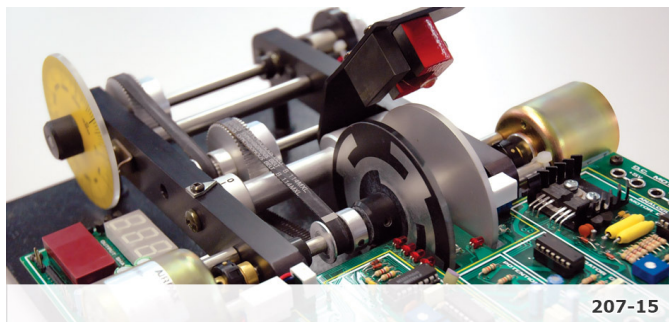
# Mechanical





## Product Information Sheet

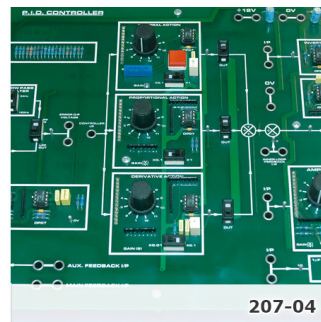
### Combined Analogue and Digital Motor Control Trainer



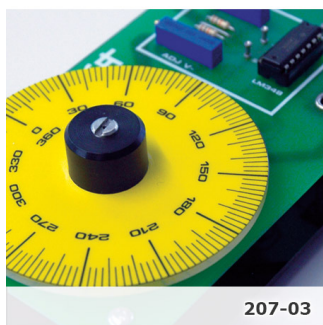
207-15



207-01



207-04



207-03



207-40



207-05



207-02

This combined teaching set provides a complete solution to teaching analog and digital motor control.

The comprehensive teaching set includes a DC motor control module, command potentiometer, PID controller module, input/output interface module, and curriculum manual. A power supply unit and connection leads are also provided.

The DC motor module is connected to a computer via a high-speed digital interface. Software is then used to provide a rich graphical environment to study the performance of speed and position control systems using both analog and digital transducers.

The mechanical unit is fitted with incremental and absolute (Gray) encoders. The motor can be controlled via a proportional voltage input or via a Pulse-Width Modulated (PWM) input. Conventional analog tachogenerator and potentiometer feedback is also provided, interfaced to the computer via analog to digital converters.

The software provides a graphical environment in which the performance of the control system can be displayed and then analyzed. The software can also be used with a simulated servo system or process plant for further study.

#### Topics include:

- The Digital Storage Oscilloscope
- Transient and Steady State Response
- Proportional Speed Control
- Proportional Position Control
- Second Order Response Parameters
- Velocity and Transient Velocity Feedback
- Controller Characteristics
- Integral Speed Control
- Proportional Plus Integral Speed Control
- Proportional Plus Integral Plus Derivative Position Control
- Instability
- Introduction to Control Systems
- Time Response
- Frequency Response
- Principles of Feedback
- Proportional Position Control
- Behavior of Second Order Systems
- Positional Control with Velocity Feedback
- Three-Term or PID Control
- Stability
- The Use of Computers for Control
- Analog Interfacing
- Direct Digital Control
- Digital Interfacing

#### Typical activities include:

- Describe the Main Elements of a PID Controller
- Investigate the data capture features of a digital storage oscilloscope.
- Conduct simple step, ramp, and frequency response tests.
- Demonstrate Proportional Speed control.
- Outline the reasons for adding velocity feedback.
- Demonstrate the change in following error when transient velocity feedback replaces velocity feedback.
- Discuss the effects of noise when subjected to derivative action.
- Demonstrate various combinations of PID control.
- Demonstrate the stabilizing effect of a proportional action on a system controlled with integral action.
- Operate the software controls to drive the DC Motor and observe its behavior.
- Measure the parameters of a plant using step tests.
- Account for the excellent steady state performance of a proportional servo position system.
- Set up a servo system to respond to ramp inputs using proportional gain and transient velocity feedback.

## LJ Create –Product Information Sheet (Continued)

### Combined Analogue and Digital Motor Control Trainer



- Show how positive feedback can occur in a negative feedback system.
- Explain Gain Margin and Phase Margin.
- Describe the advantages and disadvantages of using computers in real-time control.
- Outline the problems arising from low resolution and slow sampling.
- Describe Gray code position measurement and other absolute and relative digital position measurement techniques.

The Virtual Control Laboratory software will allow the control and monitoring of a DC motor module and command potentiometer, and it includes the following items:

- Real time Windows based Virtual
- Control Laboratory software
- CLIO Control laboratory input/output
- interface module
- Parallel interface lead for connection
- between the interface module and the PC

The software allows the user to divide the screen into four independently configurable sections:

- Control Reference signal
- Controller
- Plant
- Display

The control reference signal can be set from an internal signal generator that can provide sine, step, ramp, sawtooth, random, DC level and pulse waveforms.

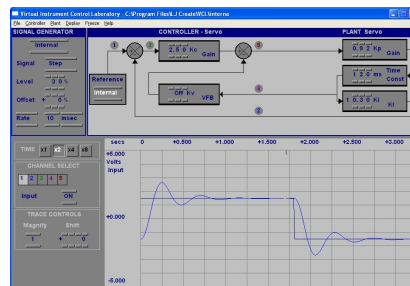
The controller is also capable of being set to open, proportional, servo and PID modes.

The output signal from the CLIO module capable of:

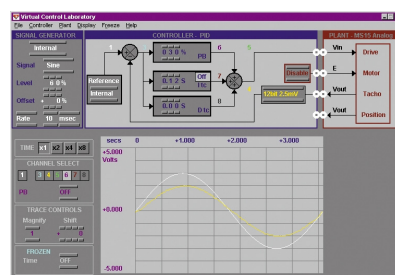
- Outputting an analog signal while measuring voltages representing motor position and velocity
- Outputting a PWM signal while measuring position and velocity using gray code and slotted disks
- Simulating a servo system by providing output and velocity signals
- Simulating a 2nd order system without an integrator

The display is able to show eight channels in real time. The channels are color-coded and can be being switched independently off if required. The display information can be frozen and exported in Excel or Matlab format, as well as a text file.

The interface module has a power requirement of +5Vat 100mA.



Step Response Example Screen



PID Control Example Screen

#### Items Included:

- 207-01 Curriculum Manual Set
- 207-02 Virtual Control Laboratory
- 207-03 Command Potentiometer
- 207-04 PID Controller Module
- 207-05 4mm Connection Leads
- 207-15 D.C. Motor Control Module
- 207-40 Power Supply

#### Other Items Required:

- Windows-based Computer
- Dual Trace Oscilloscope
- Function Generator
- Multimeter

#### General Information:

Weight: 8.5 kg

Dimensions: See individual info

Supply Voltage: 50-60Hz 110-120V

AC or 220-240V AC

Packed Volume: 0.05 m<sup>3</sup>

Packed Weight: 10.1 kg

**Order Code: 207-00**

P8458-A

**For more information email us [info@ljcreate.co.uk](mailto:info@ljcreate.co.uk) or visit [www.ljcreate.com](http://www.ljcreate.com)**